LLNL-JRNL-659526

Fission Reaction Event Yield Algorithm -- FREYA -- For Event-by -Event Simulation of Fission

J. Verbeke, J. Randrup, R. Vogt

September 2, 2014

Computer Physics Communication

Disclaimer

This document was prepared as an account of work sponsored by an agency of the United States government. Neither the United States government nor Lawrence Livermore National Security, LLC, nor any of their employees makes any warranty, expressed or implied, or assumes any legal liability or responsibility for the accuracy, completeness, or usefulness of any information, apparatus, product, or process disclosed, or represents that its use would not infringe privately owned rights. Reference herein to any specific commercial product, process, or service by trade name, trademark, manufacturer, or otherwise does not necessarily constitute or imply its endorsement, recommendation, or favoring by the United States government or Lawrence Livermore National Security, LLC. The views and opinions of authors expressed herein do not necessarily state or reflect those of the United States government or Lawrence Livermore National Security, LLC, and shall not be used for advertising or product endorsement purposes.

Fission Reaction Event Yield Algorithm
FREYA
for Event-by-Event Simulation of Fission

J. M. Verbeke[1]*, J. Randrup[2], R. Vogt[1,3]
[1]Lawrence Livermore National Laboratory, P.O. Box 808, Livermore, CA 94551, USA
[2]Lawrence Berkeley National Laboratory, 1 Cyclotron Road, Berkeley, CA 94720, USA
[3]University of California, Davis, One Shields Avenue, Davis, CA 95616, USA

Abstract

From nuclear materials accountability to detection of special nuclear material, SNM, the need for better modeling of fission has grown over the past decades. Current radiation transport codes compute average quantities with great accuracy and performance, but performance and averaging come at the price of limited interaction-by-interaction modeling. For fission applications, these codes often lack the capability of modeling interactions exactly: energy is not conserved, energies of emitted particles are uncorrelated, prompt fission neutron and photon multiplicities are uncorrelated. Many modern applications require more exclusive quantities than averages, such as the fluctuations in certain observables (e.g. the neutron multiplicity) and correlations between neutrons and photons. The new computational model, FREYA (Fission Reaction Event Yield Algorithm), aims to meet this need by modeling complete fission events. Thus it automatically includes fluctuations as well as correlations resulting from conservation of energy and momentum. FREYA has been integrated into the LLNL Fission Library, and will soon be part of MCNPX2.7.0, MCNP6, TRIPOLI-4.9, and Geant4.10.

Program summary:

Program title: FREYA 1.0
Catalogue identifier:
Program summary URL:
Program obtainable from:
Licensing provisions:
No. of lines in distributed program, including test data, etc: TBD
No. of bytes in distributed program, including test data, etc: TBD
Distribution format: tar.gz
Programming language: Fortran 90, C++
Computer: any computer with a Fortran 90 and a C++ compiler, tested with (a) Intel Xeon CPU X5660, 2.8 GHz, 48 GB RAM, (b) Intel(R) Xeon(R) CPU E5-2620, 2 GHz, 64 GB RAM, (c) Intel Xeon CPU W3520, 2.67 GHz, 6 GB RAM
Operating system: (a) Red Hat Enterprise Linux Server release 6.5 (GNU Fortran and g++ (GCC) 4.4.7 20120313 (Red Hat 4.4.7-4)), (b) CentOS release 6.4 (GNU Fortran and g++ (GCC) 4.4.7 20120313 (Red Hat 4.4.7-3)), (c) MacBook Pro OSX 10.6.8 (GNU Fortran and g++-mp-4.7 (MacPorts gcc47 4.7.3_5) 4.7.3)
Has the code been vectorized or parallelized? No

[*] email: verbeke2@llnl.gov, fax: +1 (925) 422-7310

RAM: 6 GB
Classification: 17.8
External routines: None.
Nature of problem: Modeling of fission events.
Solution method: Simulation of complete fission events, production of secondary fission fragments, fission neutrons and photons.
Restrictions: Restricted to spontaneous fission of ^{238}U, ^{240}Pu, ^{244}Cm, ^{252}Cf; neutron-induced fission of ^{233}U, ^{235}U, ^{239}Pu, for incident neutron energies less than 20 MeV.
Unusual features:
Running time: 8 s for 1M events

Keywords: fission, event-by-event Monte Carlo

1 Introduction

Several general-purpose Monte Carlo codes (MCNP/X [1–5], TART [5, 6], COG [5, 7], Geant [8–10], etc.) are currently available for modeling neutron transport. To model fission, they employ the "average fission model", which is characterized by outgoing projectiles (fission neutrons and photons) that are uncorrelated and sampled from the same probability density function. This approximation is sufficient for the calculation of average quantities such as flux, energy deposition and multiplication. However, it is unsuitable for studying detailed correlations between neutrons and/or photons on an event-by-event basis.

During the past decade several code extensions have been developed that allow the modeling of correlations in fission. MCNP-DSP [5, 11] and MCNPX-PoliMi [5, 12] added angular correlations of fission neutrons by assuming the ^{252}Cf spontaneous fission distribution can be employed for all fissionable nuclides. Both codes also include detailed multiplicity and energy distributions for prompt fission photons time-correlated with the fission event. A new option was introduced in MCNPX2.7.0 [13] for the treatment of fission events utilizing a library [14] developed at Lawrence Livermore National Laboratory (LLNL). The LLNL Fission Library features time-correlated sampling of photons from neutron-induced fission, photofission and spontaneous fission. The capabilities for correlations are, however, limited for these last 3 options (MCNP-DSP, MCNPX-PoliMi, LLNL Fission Library), as they sample outgoing particles from average fission distributions instead of sampling them from individual realizations of a fission process.

In recent years, various simulation treatments have made it possible to also address fluctuations of and correlations between fission observables. In particular, a Monte Carlo approach was developed [15, 16] for the sequential emission of neutrons and photons from individual fission fragments in binary fission. The more recent event-by-event fission model, FREYA, includes more fission isotopes and has been specifically designed for producing large numbers of fission events in a fast simulation [17–22]. Employing nuclear data for fragment-mass and kinetic-energy distributions, using statistical evaporation models for neutron and photon emission, and conserving energy, momentum, and angular momentum throughout, FREYA is able to predict a host of correlation observables, including correlations in neutron multiplicity, energy, and angles, and the energy sharing between neutrons and photons.

The stand-alone fission event generator FREYA was integrated into the LLNL Fission Library version 1.9. The LLNL Fission Library is an integral part of the transport codes MCNPX2.7.0, MCNP6, TRIPOLI-4.9 [23] (through the LLNL Fission Library version 1.9), and Geant4.10 (here the LLNL Fission Library version 1.2 is incorporated into the source code with the possibility of using version 1.9 as an external library). We are planning on distributing major upgrades of the LLNL Fission Library, with FREYA whenever available.

Basic validation is performed on FREYA whenever changes to the code are made through a suite of regression tests: the number distribution of fission neutrons is checked; the average number of neutrons and photons emitted by fission is determined to be within preset bounds; and the fission neutron and photon energies are compared to expected values.

The first part of this paper will focus on the physics in the FREYA fission model and the algorithmic implementation thereof, the second part will describe the data files required by FREYA, while the third part will show how to use FREYA within the LLNL Fission Library.

2 Fission model and algorithm

The algorithmic flow of FREYA is illustrated in Figs. 1-6. Blue boxes indicate entry points, parallelograms input parameters, cylinders data files, diamonds decision points, pentagons off page connectors, orange ovals outputs, and dice indicate sampling.

FREYA models the fission of excited nuclei. Finite excitation energies can be generated in a variety of ways, including electromagnetic agitation and, more commonly for current applications, absorption of a neutron by a fissile nucleus. In this case, the fissile nucleus ${}^{A_0}Z_0$ is in its ground state $E_0^* = 0$ before the incident neutron is absorbed, see blue box in Fig. 1; Z_0, A_0, and E_n are entered where E_n is the kinetic energy of the incoming neutron absorbed by the specified fissile nucleus ${}^{A_0}Z_0$. In its simplest version, FREYA assumes that the initial nucleus ${}^{A_0}Z_0$ is at rest, therefore $P_0 = 0$.

Depending on its degree of excitation, the system may emit one or more neutrons prior to fission, either by pre-equilibrium emission (at the highest excitations) or by (possibly sequential) pre-fission evaporation which may continue as long as the excitation energy exceeds the neutron separation energy. The two processes are described in Figs. 1-2.

In the case of spontaneous fission, the fissile nucleus ${}^{A_0}Z_0$ fissions with no prior neutron absorption, and the excitation energy of the nucleus is $E_0^* = 0$, see blue box in Fig. 3.

After the system is prepared for either neutron-induced or spontaneous fission, the fission fragments are selected by sampling, as described in Fig. 3. The scission process is finalized by determining the fission Q value, as in Fig. 4. The fission fragment kinetic and excitation energies are then determined, as shown in Fig. 5. Finally, neutron evaporation and photon emission from the fragments, as shown in Fig. 6, complete the fission event.

This section is divided into three parts to describe the physics behind the algorithms in FREYA. The system pre-fission is described in Sec. 2.1. Next, the scission process itself that results in binary fission with two excited fission fragments is treated in Sec. 2.2. Finally, de-excitation by post-fission radiation of neutrons and photons is described in Sec. 2.3.

2.1 Pre-fission

For sufficiently high incident neutron energies, there are two possible ways for neutrons to be emitted before fission occurs: pre-equilibrium neutron emission and pre-fission neutron evaporation, referred to as multichance fission. FREYA handles both these possibilities.

2.1.1 Pre-equilibrium neutron emission

Pre-equilibrium neutron emission occurs if a neutron is emitted before the compound nucleus is equilibrated. In this case, a non-equilibrium model employing a two-component exciton model that represents the evolution of the nuclear reaction in terms of time-dependent populations of ever more complex many-particle-many-hole states is used. See Ref. [20] for more details.

A many-exciton state consists of $p_{\nu(\pi)}$ neutron (proton) particle excitons and $h_{\nu(\pi)}$ neutron (proton) hole excitons. The total number of neutron (proton) excitons in the state is $n_{\nu(\pi)} = p_{\nu(\pi)} + h_{\nu(\pi)}$. Processes that reduce the number of excitons are neglected. The pre-equilibrium neutron emission spectrum is then given by

$$\frac{d\sigma_n}{dE} = \sigma_{\rm CN} \sum_{p_\pi=0}^{p_\pi^{\max}} \sum_{p_\nu=1}^{p_\nu^{\max}} W(p_\pi,h_\pi,p_\nu,h_\nu,E)\,\tau(p_\pi,h_\pi,p_\nu,h_\nu)\,P(p_\pi,h_\pi,p_\nu,h_\nu) \tag{1}$$

where $\sigma_{\rm CN}$ is the compound nuclear cross section (usually obtained from an optical model calculation), W is the rate for emitting a neutron with energy E from the exciton state $(p_\pi,h_\pi,p_\nu,h_\nu)$, τ is the lifetime of this state, and $P(p_\pi,h_\pi,p_\nu,h_\nu)$ is the (time-averaged) probability for the system to survive the previous stages and arrive at the specified exciton state. In the two-component model, contributions to the survival probability from both particle creation and charge exchange need to be accounted for. The survival probability for the exciton state $(p_\pi,h_\pi,p_\nu,h_\nu)$ can be obtained from a recursion relation starting from the initial condition $P(p_\nu=1,h_\nu=0,p_\pi=0,h_\pi=0)=1$ and setting $P=0$ for terms with negative exciton number.

For each event generated, FREYA first considers the possibility of pre-equilibrium neutron emission and, if it occurs, a neutron is emitted with an energy selected from the calculated pre-equilibrium spectrum, Eq. (1). The possibility of equilibrium neutron evaporation is then considered, starting either from the original compound nucleus, e.g. ^{240}Pu* for ^{239}Pu(n,f), or the less excited nucleus, ^{239}Pu*, remaining after pre-equilibrium emission has occurred. Neutron evaporation continues until the excitation energy of a daughter nucleus is below the fission barrier (in which case the event is abandoned and a new event is generated) or the nucleus fissions.

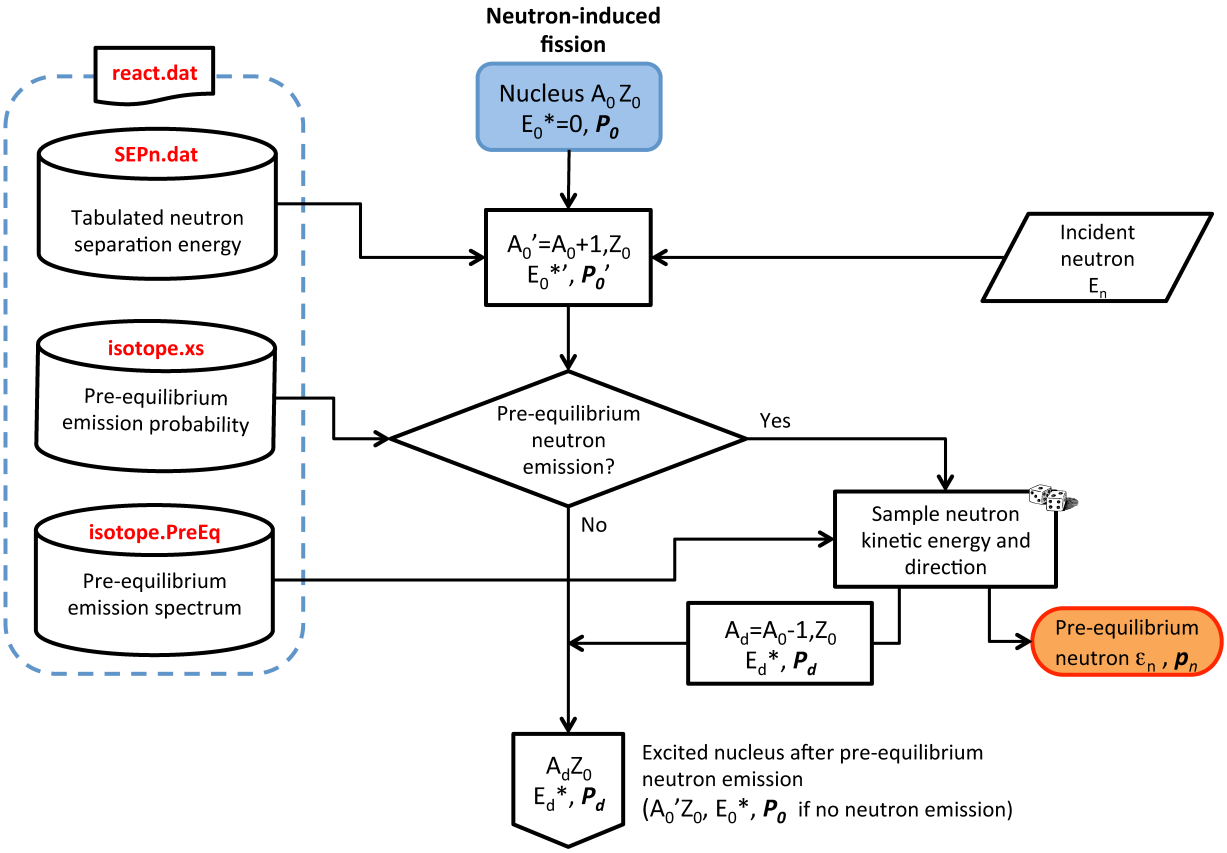

Neutron-induced fission
Nucleus $A_0 Z_0$
E_0*=0, $\boldsymbol{P_0}$
react.dat
SEPn.dat
Tabulated neutron separation energy
isotope.xs
Pre-equilibrium emission probability
isotope.PreEq
Pre-equilibrium emission spectrum
A_0'=A_0+1,Z_0
E_0*', $\boldsymbol{P_0}$'
Incident neutron E_n
Pre-equilibrium neutron emission?
Yes
No
Sample neutron kinetic energy and direction
A_d=A_0-1,Z_0
E_d*, $\boldsymbol{P_d}$
Pre-equilibrium neutron ε_n , $\boldsymbol{p_n}$
$A_d Z_0$
E_d*, $\boldsymbol{P_d}$
Excited nucleus after pre-equilibrium neutron emission
(A_0'Z_0, E_0*, $\boldsymbol{P_0}$ if no neutron emission)

Figure 1: Pre-equilibrium neutron emission, associated with neutron-induced fission.

A flow chart for pre-equilibrium neutron emission is shown in Fig. 1. One should note that pre-equilibrium neutron emission is very improbable, on average 0.10 pre-equilibrium neutrons are emitted at $E_n \sim 14$ MeV, see Fig. 2 in Ref. [20]. After this first process, the excited nucleus can also undergo pre-fission neutron evaporation, discussed next.

2.1.2 Pre-fission neutron evaporation

In multichance fission or pre-fission neutron evaporation — shown in Fig. 2 — neutron evaporation can occur from the compound nucleus as long as the excitation energy of the compound exceeds the neutron separation energy, S_n. One or more neutrons can be emitted before fission. The probability for pre-fission neutron evaporation is determined by the competition between fission and neutron evaporation. At higher incident neutron energies, neutron evaporation from the compound nucleus competes more favorably with direct (first chance) fission.

The criterion for pre-fission evaporation is based on $\Gamma_n(E^*)/\Gamma_{\rm f}(E^*)$, the relative magnitudes of the neutron

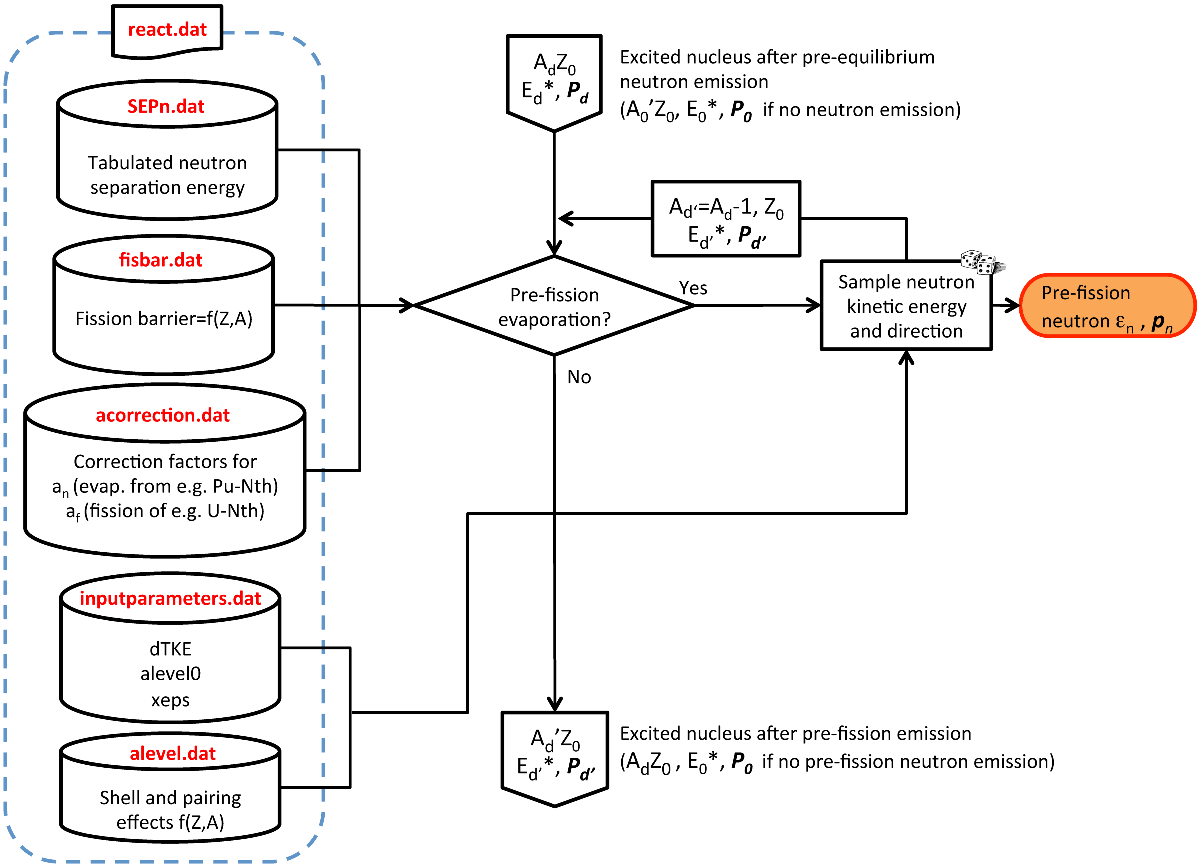

react.dat
SEPn.dat
Tabulated neutron separation energy
fisbar.dat
Fission barrier=f(Z,A)
acorrection.dat
Correction factors for
a$_n$ (evap. from e.g. Pu-Nth)
a$_f$ (fission of e.g. U-Nth)
inputparameters.dat
dTKE
alevel0
xeps
alevel.dat
Shell and pairing effects f(Z,A)
A$_d$Z$_0$ E$_d$*, P_d
Excited nucleus after pre-equilibrium neutron emission
(A$_0$'Z$_0$, E$_0$*, P_0 if no neutron emission)
A$_{d'}$=A$_d$-1, Z$_0$ E$_{d'}$*, $P_{d'}$
Pre-fission evaporation?
Yes
No
Sample neutron kinetic energy and direction
Pre-fission neutron ε$_n$, p_n
A$_d$'Z$_0$ E$_{d'}$*, $P_{d'}$
Excited nucleus after pre-fission emission
(A$_d$Z$_0$, E$_0$*, P_0 if no pre-fission neutron emission)

Figure 2: Pre-fission evaporation, associated with neutron-induced fission.

evaporation and fission decay widths at a given excitation energy E^* [24],

$$\frac{\Gamma_n(E^*)}{\Gamma_{\rm f}(E^*)} = \frac{2g_{\rm n}\mu_n\sigma}{\pi\hbar^2}\frac{\int_0^{X_n}(X_n-x)\rho_n(x)dx}{\int_0^{X_{\rm f}}\rho_{\rm f}(x)dx} , \tag{2}$$

where $g_{\rm n}=2$ is the spin degeneracy of the neutron, μ_n is its reduced mass, and $\sigma=\pi R^2=\pi r_0^2 A^{2/3}$. Here $\rho_n(x)$ is the level density in the evaporation daughter nucleus at excitation energy x. The maximum value of x is given $X_n=Q_n=E^*-S_n$, where Q_n is the Q value for neutron emission and S_n is the neutron separation energy. Similarly, $\rho_{\rm f}(x)$ is the level density of the fissioning nucleus when its shape is that associated with the top of the fission barrier. The excitation x is measured relative to the barrier top with a maximum value of $X_{\rm f}=E^*-B_{\rm f}$, where $B_{\rm f}$ is the height of the fission barrier.

Neutron evaporation is possible whenever the excitation energy of the compound nucleus is larger than the neutron separation energy, $E^*>S_n$, a positive quantity because it costs energy to remove a neutron from the nucleus. The excitation energy of the evaporation daughter nucleus is $E_d^*=E^*-S_n-E$ where E is the kinetic energy of the relative motion between the emitted neutron and the daughter nucleus. If this quantity exceeds the fission barrier in the daughter nucleus, then second-chance fission is possible. The same procedure is then applied to the daughter nucleus, so that further pre-fission neutron emission possible. As the incident neutron energy is increased, emission of further pre-fission neutrons becomes possible and the associated fission events may be classified as first-chance fission (when there are no pre-fission neutrons emitted), second-chance fission (when one neutron is emitted prior to fission), and so on. See Fig. 1 of Ref. [20] for a plot of the multichance fission probability up to $E_n = 20$ MeV for ^{239}Pu(n,f).

Emitted neutron kinetic energies are sampled using an algorithm similar to neutron evaporation explained in Sec. 2.3.1. For more discussion, see Ref. [20].

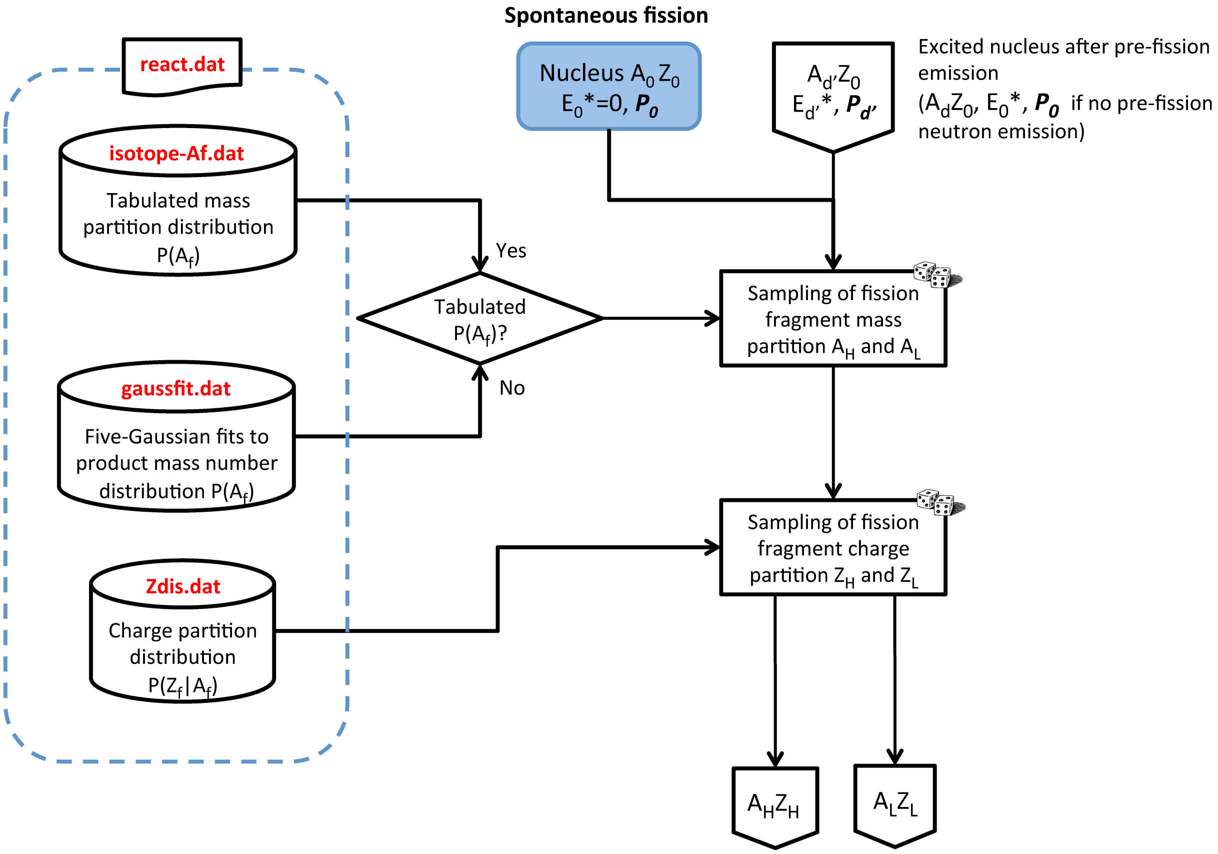
After both pre-equilibrium neutron emission and pre-fission neutron evaporation, the excitation energy is adjusted and the energy at which the yields are sampled is reduced accordingly.

2.2 Fission

After pre-fission radiation, the physics issue concerns how the mass and charge of the initial compound nucleus is partitioned among the two fission fragments and how the available energy is divided between the excitation of the two fragments and their relative kinetic energy.

2.2.1 Fission fragment mass and charge distributions

The current understanding of the fission process is that the evolution from the initial compound nucleus to two distinct fission fragments occurs gradually as a result of a dissipative multidimensional evolution of the nuclear shape. Because no quantitatively reliable theory has yet been developed for this process, we employ empirical evidence as a basis for selecting the mass and charge partition.



Spontaneous fission
react.dat
Nucleus $A_0 Z_0$
E_0*=0, P_0
$A_{d'}Z_0$
$E_{d'}$*, $P_{d'}$
Excited nucleus after pre-fission emission
(A_dZ_0, E_0*, P_0 if no pre-fission neutron emission)
isotope-Af.dat
Tabulated mass partition distribution P(A_f)
Yes
Tabulated P(A_f)?
No
Sampling of fission fragment mass partition A_H and A_L
gaussfit.dat
Five-Gaussian fits to product mass number distribution P(A_f)
Zdis.dat
Charge partition distribution P(Z_f|A_f)
Sampling of fission fragment charge partition Z_H and Z_L
A_HZ_H
A_LZ_L

Figure 3: Selection of fission fragments, applicable both to neutron-induced and spontaneous fission.

The compound nucleus left after pre-fission radiation undergoes binary fission into a heavy $^{A_H}Z_H$ and a light fragment ${}^{A_L}Z_L$, see Fig. 3. The fragment masses are obtained from experimental mass yields $Y(A_f)$ — see Refs. [17, 19] — in the case of spontaneous fission. For neutron-induced fission, the energy dependence of $Y(A_f)$ has been modeled for incident neutron energies up to 20 MeV [20]. When data are not available for the mass yields $Y(A_f)$, the mass number A_f of one of the fission fragments is selected randomly from a probability density $Y(A_f)$ for which we employ five-Gaussian fits to the product mass number distribution [25] shifted upward in mass to ensure a symmetric distribution of the primary fragments. Sec. 3.7 covers the five-Gaussian fits in more detail.

The charge partition is selected subsequently from the associated conditional probability distribution $P(Z_f|A_f)$. For this, we follow Ref. [26] and employ a Gaussian form

$$P(Z_f|A_f) \propto \exp\left(-\frac{(Z_f - \overline{Z}_f(A_f))^2}{2\sigma_Z^2}\right) \tag{3}$$

with the condition $\left|Z_f - \overline{Z}_f(A_f)\right| \leq 5\sigma_Z$. The centroid is determined by requiring that the fragments have, on average, the same charge-to-mass ratio as the fissioning nucleus $\overline{Z}_f(A_f) = A_f(Z_0/A_0)$. The charge of the complementary fragment then follows using $Z_L + Z_H = Z_0$.

2.2.2 Fragment energies

Once the mass and charge of the two fragments have been selected, the Q value of the fission channel is the difference between the total mass of A_0 and the fragment ground-state masses,

$$Q_{LH} = M(A_0) - M_L - M_H\ . \tag{4}$$

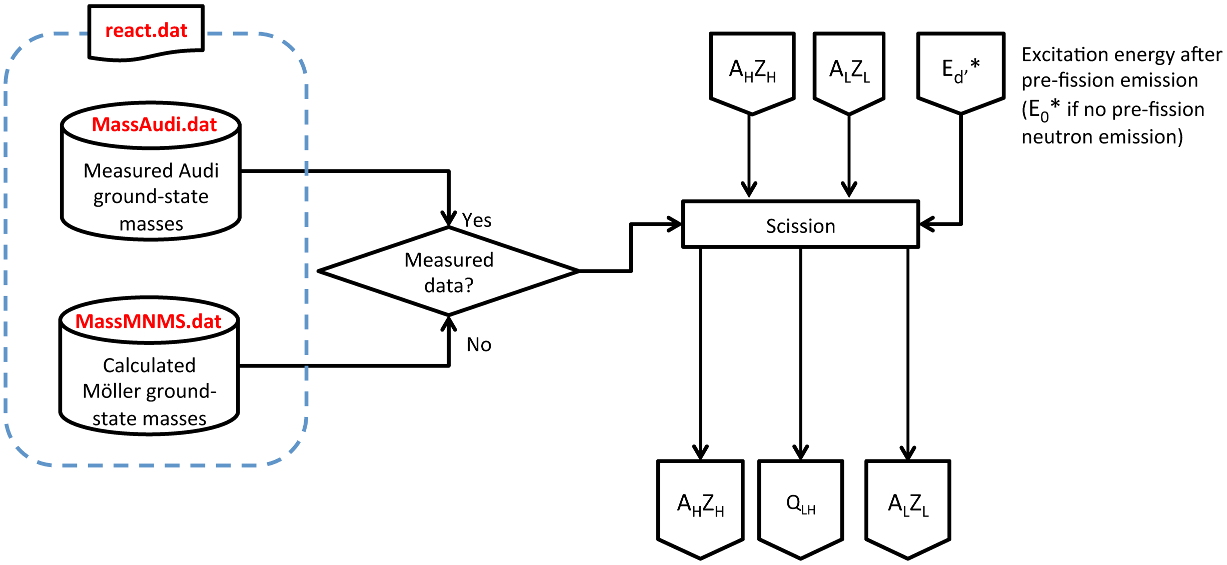

react.dat
MassAudi.dat
Measured Audi ground-state masses
MassMNMS.dat
Calculated Möller ground-state masses
Yes
Measured data?
No
A_HZ_H
A_LZ_L
$E_{d'}*$
Excitation energy after pre-fission emission (E_0* if no pre-fission neutron emission)
Scission
A_HZ_H
Q_{LH}
A_LZ_L

Figure 4: Scission, applicable both to neutron-induced and spontaneous fission.

FREYA takes the required nuclear ground-state masses from the compilation by Audi and Wapstra [27], supplemented by the calculated masses of Möller *et al.* [28] when no data are available. This simple process is shown in Fig. 4.

Figure 5 shows how the fission fragment energies are selected. The Q_{LH} value is divided between the total kinetic energy (TKE) and the total excitation energy (TXE) of the fragments. The average TKE is assumed to

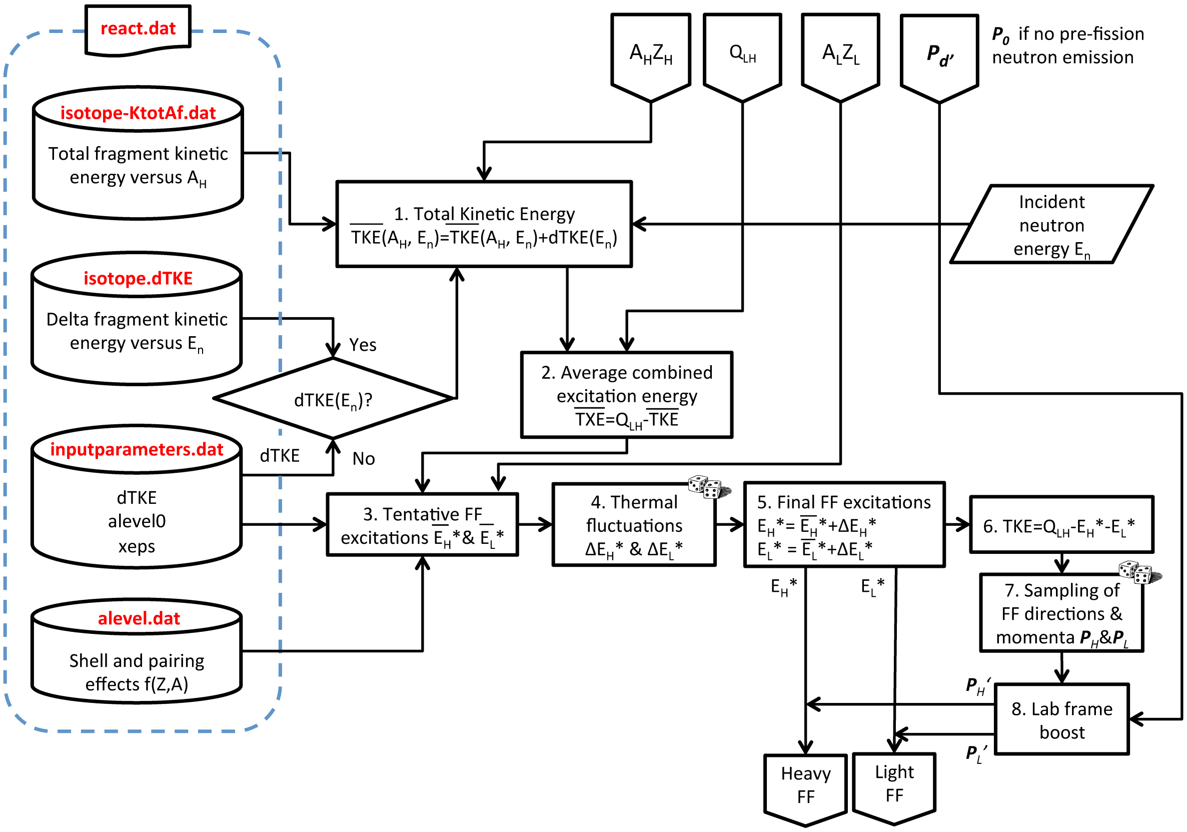

react.dat
isotope-KtotAf.dat
Total fragment kinetic energy versus A_H
isotope.dTKE
Delta fragment kinetic energy versus E_n
inputparameters.dat
dTKE
alevel0
xeps
alevel.dat
Shell and pairing effects f(Z,A)
A_HZ_H
Q_{LH}
A_LZ_L
$P_{d'}$
P_0 if no pre-fission neutron emission
1. Total Kinetic Energy
$\overline{\mathrm{TKE}}(A_H, E_n)=\overline{\mathrm{TKE}}(A_H, E_n)+\mathrm{dTKE}(E_n)$
Incident neutron energy E_n
Yes
$\mathrm{dTKE}(E_n)$?
No
dTKE
2. Average combined excitation energy
$\overline{\mathrm{TXE}}=Q_{LH}-\overline{\mathrm{TKE}}$
3. Tentative FF excitations $\overline{E}_H$*& $\overline{E}_L$*
4. Thermal fluctuations ΔE_H* & ΔE_L*
5. Final FF excitations
E_H*= $\overline{E}_H$*+ΔE_H*
E_L* = $\overline{E}_L$*+ΔE_L*
6. TKE=Q_{LH}-E_H*-E_L*
7. Sampling of FF directions & momenta P_H&P_L
E_H*
E_L*
P_H'
P_L'
8. Lab frame boost
Heavy FF
Light FF

Figure 5: Selection of fission fragment energies.

take the form

$$\overline{\mathrm{TKE}}(A_H,E_n) = \overline{\mathrm{TKE}}_{\mathrm{data}}(A_H) + d\mathrm{TKE}(E_n)\,. \tag{5}$$

The first term is extracted from data on thermal neutrons while the second is adjusted to the measured average neutron multiplicity, $\overline{\nu}$.

After the average total fragment kinetic energy, $\overline{\mathrm{TKE}}$, has been calculated, the combined statistical fragment excitation energy, $\overline{\mathrm{TXE}}$, follows from energy conservation,

$$\begin{aligned}\overline{\mathrm{TXE}} &= \overline{E}^*_L + \overline{E}^*_H \\ &\doteq Q_{LH} - \overline{\mathrm{TKE}}\,.\end{aligned} \tag{6}$$

The first relation indicates that the TXE is partitioned between the two fragments. If the fragments are in mutual thermal equilibrium, as might be expected since they are in thermal contact prior to scission, their temperatures are equal, $T_L = T_H$. The code tentatively assumes that $\overline{\mathrm{TXE}}$ is divided into fragment excitation energies $\tilde{E}^*_f$ in the following proportions: $\tilde{E}^*_f = (A_f/A_0)\overline{\mathrm{TXE}}$. The $\tilde{E}^*_f$'s are then used to refine the excitation energies using

$$\acute{E}^*_f = \frac{a_f(\tilde{E}^*_f)}{a_L(\tilde{E}^*_L)+a_H(\tilde{E}^*_H)}\overline{\mathrm{TXE}}\,. \tag{7}$$

where $a_f(E_f)$ is the level density parameter for fragment f. It depends on the fragment excitation energy. To account for the microscopic structure of the individual fragments as well as any possible energy dependence, FREYA uses the functional form due to Kawano [29],

$$a_f(\tilde{E}_f^*) = \frac{A_f}{e_0}\left[1 + \frac{\delta W_f}{U_f}\left(1 - e^{-\gamma U_f}\right)\right] , \tag{8}$$

where $\gamma = 0.05$ [15] and $U_f = E_f^* - \Delta_f$. U_f is the statistical part (the "heat") of the excitation energy. The pairing energy of the fragment, Δ_f, and its shell correction, δW_i are those calculated by Koura *et al.* [30] for nuclei with $20 \leq Z_f \leq 92$. Eq. (8) depends on the asymptotic level density parameter e_0[1], which is adjustable on a per nucleus basis, but FREYA currently assumes a universal value of 10.0724 MeV for e_0 from the fit in Ref. [20].

Because the observed neutron multiplicities suggest that the light fragments are more excited (probably due to their greater distortion at scission), the average excitations are adjusted as

$$\overline{E}_L^* = x\acute{E}_L^* , \tag{9}$$

$$\overline{E}_H^* = \overline{\mathrm{TXE}} - \overline{E}_L^* , \tag{10}$$

where $x > 1$ is a parameter[2] adjusted for each fissile nucleus based on the sawtooth shape of the neutron multiplicity as a function of fragment mass, see Ref. [20].

After the mean excitation energies have been assigned, FREYA accounts for thermal fluctuations. The fragment temperature T_f is obtained from $\overline{U}_f \equiv U_f(\overline{E}_f^*) = a_f T_f^2$, where $U(E^*) = E^*$. The variance in the excitation E_f^* is then $\sigma_f^2 = 2\overline{U}_f^* T_f$.

Therefore, for each of the two fragments, we sample a thermal fluctuation δE_f^* from a normal distribution of variance σ_f^2 and modify the fragment excitation energies as, $E_f^* = \overline{E}_f^* + \delta E_f^*$. Energy conservation causes a compensating fluctuation in TKE leading to $\mathrm{TKE} = \overline{\mathrm{TKE}} - \delta E_L^* - \delta E_H^*$ [20].

The fragment momenta after scission, in the center of mass frame of the compound nucleus, are equal and opposite with a magnitude determined from TKE and the fragment rest energies. The azimuthal and polar angles of the light fragment are randomly sampled and a spherical unit vector is constructed from these angles to be the original direction of the light fragment. The momentum vector is then the magnitude of the momentum multiplied by the previously constructed unit vector. The direction of the heavy fragment is, by definition, equal and opposite to that of the light fragment. These center of mass momenta are subsequently Lorentz-boosted to the laboratory frame.

2.3 Post-fission radiation

FREYA assumes that the fully accelerated fission fragments first de-excite by sequential neutron evaporation, followed by sequential photon emission, see Fig. 6.

2.3.1 Neutron evaporation

Neutron evaporation occurs after the fragments have reached their asymptotic velocities. We treat post-fission neutron radiation by iterating a simple treatment of single neutron evaporation until no further neutron emission is energetically possible.

A fission fragment is an excited nucleus with a total mass equal to its ground-state mass plus its excitation energy, $M_f^* = M_f^{\mathrm{gs}} + E_f^*$. The Q value for neutron emission is then

$$\begin{aligned} Q_n &= M_f^* - M_d^{\mathrm{gs}} - m_n \\ &= M_f^{\mathrm{gs}} + E_f^* - M_d^{\mathrm{gs}} - m_n \end{aligned} \tag{11}$$

[1] alevel0 in file 'inputparameters.dat'.

[2] xeps in file 'inputparameters.dat'.

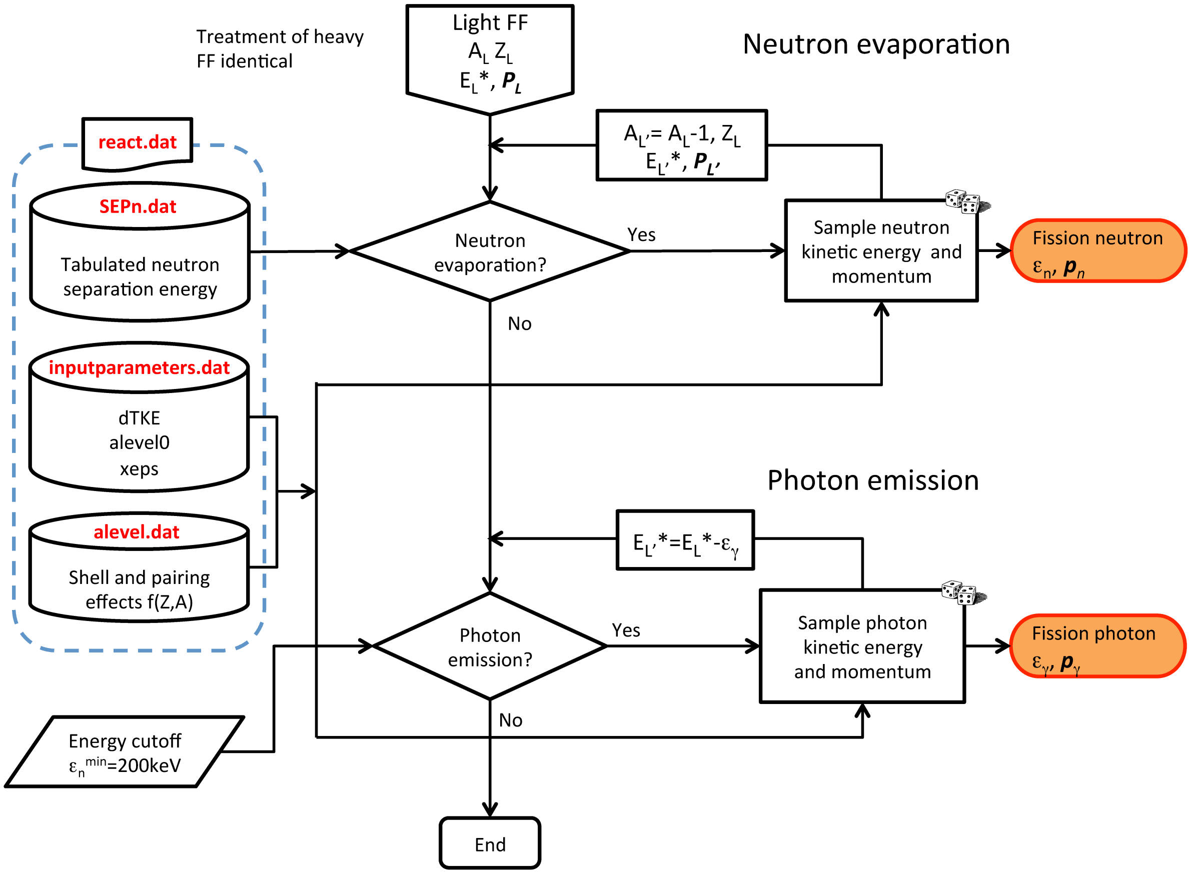


Figure 6: Neutron evaporation and photon emission.

where $M_d^{\rm gs}$ is the ground-state mass of the daughter nucleus, and m_n is the mass of the ejectile. Using the definition for the neutron separation energy $S_n(Z,A) = -M\left({}^{A}Z\right) + M\left({}^{A-1}Z\right) + m_n$, we have

$$Q_n = E_f^* - S_n(Z,A) \ . \tag{12}$$

The Q value equals the maximum possible excitation energy of the daughter nucleus $Q_n = E_f^{\rm max}$ for vanishing final relative kinetic energy of the ejectile, or when the emitted neutron has no kinetic energy.

Once the Q value is known, it is straightforward to sample the kinetic energy of an evaporated neutron, assuming that it is isotropic in the rest frame of the emitting nucleus. The maximum temperature of the evaporation daughter, $T_{\rm max}$, is obtained from

$$a_d T_{\rm max}^2 = Q_n \ , \tag{13}$$

where a_d is the level-density parameter of the daughter nucleus. The neutron kinetic energy ε_n is sampled from

$$f_n(\varepsilon_n) \sim \ \varepsilon_n \exp(-\varepsilon_n/T_{\rm max}) \ . \tag{14}$$

The excitation energy of the daughter nucleus is then given by

$$E_d^* = Q_n - \varepsilon_n \ . \tag{15}$$

This procedure may be repeated as long as neutron emission is energetically possible, which happens when $E_f^* \geq S_n$, or optionally, as long as the Q_n value for emission exceeds $E_{n\,\rm cut}$, a cutoff where photon emission dominates over neutron emission. Afterwards, photon emission takes over.

2.3.2 Photon emission

After neutron evaporation has ceased, the residual product nucleus has a statistical excitation energy of $E^* < S_n(Z,A) + E_{n\mathrm{cut}}$ and de-excites by sequential statistical photon emission. Statistical photon emission is treated similarly to neutron evaporation except there is no separation energy for photons. Since the photons are massless, we introduce an infrared cut-off energy. Furthermore, there is an extra energy factor in the photon phase space,

$$f_\gamma(E) \sim E^2 \exp(-E/T) , \tag{16}$$

where T, the nuclear temperature prior to emission, is equal to the maximum possible temperature after emission. Photons are emitted isotropically in the frame of the emitter nucleus. Emission continues until the available statistical excitation energy has been exhausted.

3 FREYA data files

The data files shown in this section are for the most part organized in order of use in the general FREYA sequence of events. Their list is given in Table 1.

Table 1: List of all data files needed by FREYA and the section of the text where they are described. The label *isotope* is a placeholder for the exact compound isotope (just before fission) included, e.g. U236 for the reaction $^{235}\mathrm{U}(n,f)$. The $*$ indicates that the inputs for some of the isotopes are from multiple sources. In this case, the $*$ represents the author's name(s).

File name	Description	Section Number
react.dat	List of isotopes treated	3.1
SEPn.dat	Neutron separation energy	3.2
fisbar.dat	Nuclei fission barriers	3.3
acorrection.dat	Neutron separation energy	3.4
isotope.xs	Pre-equilibrium emission probability	3.5.1
isotope.PreEq	Pre-equilibrium emission spectra	3.5.2
Zdis.dat	Charge distribution width	3.6
isotope-Af*.dat	Fission fragment yields	3.7.1
gaussfit.dat	Five-Gaussian fit parameters	3.7.2
MassMNMS.dat	Theoretical isotopic mass tables	3.8.1
MassAudi.dat	Experimental isotope mass tables	3.8.2
isotope-KtotAf*.dat	Total kinetic energy	3.9
inputparameters.dat	Fission fragment total excitation energy and partition parameters	3.10
alevel.dat	Shell corrections and pairing effects	3.11

Upon startup, FREYA reads in a master data file containing

(1) the ZA of the available compound nuclei before fission. There are currently 7 fissionable isotopes: 4 spontaneously fissioning (^{238}U, ^{240}Pu, ^{244}Cm, ^{252}Cf) and 3 neutron-induced (^{233}U, ^{235}U, ^{239}Pu);

(2) the maximum number of pre-fission neutrons in pre-fission neutron evaporation;

(3) the names of the FREYA data files containing:

(a) the probability distributions of mass partition $P(A_f)$;

(b) the pre-equilibrium emission probabilities;

(c) the pre-equilibrium emission spectra;

(d) the kinetic energy distributions of the fission fragments.

Because additional isotopes are expected to be regularly added in the future, the code was designed to ease the extension to additional isotopes: algorithm and data are completely separated and isotopes can easily be added by adding lines to the master data file, and generating some of the required files, 3a through 3d, listed above.

3.1 Master data file 'react.dat'

The master file 'react.dat' contains all of the data files that FREYA uses. It is thus shown in Figs. 1-6. Each line in this file corresponds to a compound isotope before fission. Except for the first header line, the overall structure of the master data file is described in Table 2. For neutron-induced fission, one should point out that '*A*' is the number of nucleons in the compound nucleus, i.e. the nuclear mass number after the incident neutron has been captured by the fissile isotope. Thus the first line corresponds to the neutron-induced fission reaction $n+^{233}$U. (We use the *A* value of the compound nucleus to better distinguish neutron-induced fission from spontaneous and/or photofission. Thus the given *A* reflects the mass of the nucleus that actually fissions.) The current version of this file is reproduced below:

```
element  Z   A  reaction  max # pre-fiss n  P(Af) file name  Pre-eq prob  Pre-eq spect  # TKE files  TKE file names
U       92 234   '(n,f)'                 0    U234-Af-N.dat            -             -            1  U234-KtotAf-N.dat
U       92 236   '(n,f)'                 3                -     U-236.xs   U-236.PreEq            1  U236-KtotAf-N.dat
U       92 238      'sf'                 0      U238-Af.dat            -             -            1  U238-KtotAf.dat
Pu      94 240      'sf'                 0     Pu240-Af.dat            -             -            1  Pu240sf-Ktot.dat
Pu      94 240   '(n,f)'                 3                -     Pu240.xs   Pu240.PreEq            3  Pu240-KtotAf-N.dat Pu240-KtotAf-T.dat Pu240-KtotAf-W.dat
Cm      96 244      'sf'                 0     Cm244-Af.dat            -             -            1  Cm244-KtotAf-SH.dat
Cf      98 252      'sf'                 0   Cf252-Af-H.dat            -             -            1  Cf252-KtotAf-H.dat
```

If this file does not exist, the FREYA fission sampler outputs an error and returns. If a filename specified in 'react.dat' does not exist, FREYA outputs an error and returns. When a hyphen '-' replaces a filename, FREYA uses a default treatment, as described later.

Table 2: Structure of master data file 'react.dat'.

Column	Description
element	Symbol of the isotope in the periodic table
Z	Proton number (nuclear charge) of the element
A	Mass number (neutrons + protons) of the *compound* nucleus
reaction	Reaction type, either neutron-induced fission '(n,f)' or spontaneous fission 'sf'
max # pre-fiss n	Maximum number of pre-fission neutrons that this isotope can emit (second, third and fourth chance fission)
$P(A_f)$ file name	Name of the file containing the fission fragment yield probability $P(A_f)$
Pre-eq prob	Name of the file containing the pre-equilibrium emission probability
Pre-eq spect	Name of the file containing the pre-equilibrium emission spectrum
# TKE files	Number of total kinetic energy 'TKE' files that will be read by FREYA
TKE file names	Names of the total kinetic energy files

3.2 Neutron separation energy in file ‘SEPn.dat’

As explained in Sec. 2, the neutron separation energy is used for multiple purposes. It is used to calculate the excitation energy of the nucleus after absorption of the incident neutron (see Fig. 1) for neutron-induced fission. It determines whether pre-fission evaporation (see Fig. 2) is possible: as long as the excitation energy of the compound nucleus exceeds the neutron separation energy S_n, one or more neutrons can be emitted before fission. It is used to determine the kinetic energy of the evaporation neutrons (see Fig. 6). For a fragment of statistical excitation E^*, the maximum temperature in its evaporation daughter, T_{max}, is obtained from $aT_{\text{max}}^2 = E^* - S_n(Z,A)$. The neutron kinetic energy ε is sampled from $f_n(\varepsilon) \sim \ \varepsilon \exp(-\varepsilon/T_{\text{max}})$. Finally, it is used to calculate how many prompt neutrons are emitted by the fission fragments (see Fig. 6). Neutron evaporation ceases when the statistical excitation energy $E^* < S_n(Z,A) + E_{n\,\text{cut}}\,(Z,A)$, i.e. neutrons are emitted as long as the Q value for emission exceeds $E_{n\,\text{cut}}\,(Z,A)$, at which point photon emission takes over.

This file contains the neutron separation energy for different fissile isotopes and is used to decide whether pre-fission neutron evaporation takes place. An excerpt of the data file ‘SEPn.dat’ is shown below:

```
 Z    A   separation energy [keV]
92  218            8847.36
92  219            6782.61
(...)
98  255            4603.
98  256            5841.
```

Except for the first header line, the overall structure of this data file is explained in Table 3.

Table 3: Structure of data file ‘SEPn.dat’.

Column	Description
Z	Proton number
A	Mass number
separation energy	Neutron separation energy in units of keV

3.3 Data file ‘fisbar.dat’

The data file ‘fisbar.dat’ is used in Fig. 2 showing pre-fission evaporation, and contains fission barriers of nuclei. The role of fission barriers is explained in Sec. 2.1.2. If a nucleus is not in the table, a default fission barrier of 0 MeV is used. In this case, the upper limit of the integral in the denominator of Eq. (2), X_{f}, is just E^*.

Fission barriers are used to compute the decay width of fission. The criterion for pre-fission evaporation is not whether it is energetically possible ($E^* > S_n$), as in neutron evaporation explained in Sec. 2.3.1. Rather, a choice is made based on $\Gamma_n(E^*)/\Gamma_{\text{f}}(E^*)$, the relative magnitudes of the decay widths. Of course, $\Gamma_n(E^*)$ vanishes for E^* below S_n.

An excerpt of data file ‘fisbar.dat’ is shown below:

```
 Z   A  fiss. barrier [MeV]
92 231          5.5
92 232          5.4
(...)
94 245          5.85
94 246          5.4
```

Except for the first header line, the structure of this data file is explained in Table 4.

Table 4: Structure of data file ‘fisbar.dat’.

Column	Description
Z	proton number
A	mass number
fiss. barrier	fission barrier in MeV

3.4 Data file ‘acorrection.dat’

Similarly to data file ‘fisbar.dat’, ‘acorrection.dat’ is used in Fig. 2 for pre-fission evaporation and contains correction factors for the level-density parameters of a few isotopes of uranium and plutonium.

As is common, FREYA assumes nuclei level densities to be of the form $\rho\,(E_i^*) \sim \exp(2\sqrt{a_i U_i})$, where U_i is the effective statistical energy and a_i is the level-density parameter. Level-density parameters enter in the calculation of $\Gamma_n(E_i^*)$ and $\Gamma_\mathrm{f}(E_i^*)$, the neutron evaporation and fission decay widths [24]. For a few isotopes of uranium and plutonium, the level-density parameters a_i are multiplied by correction factors a_n and a_f for these calculations.

The data file ‘acorrection.dat’ is reproduced below in its entirety:

```
 Z   A  an correction factor  af correction factor
92 236                1.03                 0.96
92 235                1.14                 1.20
92 234                1.00                 1.00
94 240                1.00                 1.00
94 239                0.92                 1.00
94 238                1.00                 1.00
```

Except for the first header line, the overall structure of the data file ‘acorrection.dat’ is explained in Table 5.

Table 5: Structure of data file ‘acorrection.dat’.

Column	Description
Z	Proton number
A	Mass number
a_n	Correction to level-density parameter used to compute neutron evaporation decay width, Γ_n
a_f	Correction to level-density parameter used to compute fission decay width, Γ_f

3.5 Data files associated with pre-equilibrium neutron emission

In pre-equilibrium neutron emission, an incident neutron interacts with the fissile nucleus and is re-emitted afterwards before the nucleus fissions. It is different from multichance fission, where a neutron other than the incident neutron is emitted before fission. Files with pre-equilibrium emission probabilities and pre-equilibrium energy spectra are used in the algorithm for pre-equilibrium neutron emission, see Fig. 1.

In file ‘react.dat’ both ^{236}U* and ^{240}Pu* have pre-equilibrium emission probabilities and energy spectra. For a given isotope, both the pre-equilibrium emission probability and the energy spectrum are necessary. If one is given and the other is missing, the FREYA fission sampler will return with an error message.

3.5.1 Pre-equilibrium emission probabilities

The excerpt of file ‘U-236.xs’ below shows the probability of pre-equilibrium neutron emission from the compound nucleus ^{236}U as a function of the incident neutron energy E_n:

```
 U-236.xs: Probability for pre-equil emission vs En:
  100   0.20000
    0   0.00000 0.0000000E+00
    1   0.20000 0.0000000E+00
    2   0.40000 0.0000000E+00
    3   0.60000 0.0000000E+00
    4   0.80000 0.0000000E+00
    5   1.00000 0.6432276E-04
    6   1.20000 0.2734767E-03
    7   1.40000 0.6399593E-03
(...)
   99  19.80000 0.2314241E+00
  100  20.00000 0.2336182E+00
```

The overall structure of the data file, from line 3 on, is described in Table 6. The first two lines are header lines. Line 1 is an identifying comment. Starting on line 3, data entries are indexed from zero and the index of the last entry in the list is provided as the first entry on line 2. The second entry on line 2 is the width of the incident neutron energy bins in units of MeV. (The bin width is 0.2 MeV in this file.)

Table 6: Structure of the pre-equilibrium emission probability data file.

Column	Description
1	Entry number
2	Incident neutron energy, E_n, in units of MeV
3	Probability of pre-equilibrium neutron emission for incident energy E_n, $P(p_\pi, h_\pi, p_\nu, h_\nu)$ in Sec. 2.1.1

For the compound nucleus ^{236}U, we read in file ‘U-236.xs’ that the probability for pre-equilibrium neutron emission is null for neutron incident energies from 0 to 0.8 MeV.

3.5.2 Pre-equilibrium emission spectra

An example of the neutron spectrum from pre-equilibrium emission is shown below as a function of E_n. These lines are taken from file ‘U-236.PreEq’ for the compound nucleus ^{236}U:

```
Pre-equilibrium neutron spectra from U236* for various En (.PreEq):
  100 incident neutron energies En in steps of   0.200 MeV:
    0   0.000    0  0.199153
    1   0.200    0  0.199153
    2   0.400    0  0.199153
    3   0.600    0  0.199153
    4   0.800    0  0.199153
    5   1.000    1  0.199153
  0.199153 0.2510638E+01
    6   1.200    2  0.199153
  0.199153 0.3564677E+01
  0.398305 0.1456601E+01
```

```
    7   1.400    3  0.199153
  0.199153 0.1674361E+01
  0.398305 0.2424930E+01
  0.597458 0.9219841E+00
(...)
  100  20.000   96  0.199153
  0.199153 0.1114254E-01
  0.398305 0.2194049E-01
  0.597458 0.2977476E-01
(...)
 18.720339 0.6576045E-02
 18.919493 0.4754821E-02
 19.118645 0.1891974E-02
```

Except for the first two header lines, the overall structure of the data file for the pre-equilibrium neutron emission spectrum is explained in Tables 7 and 8. The structure description in Table 7 applies to lines 3-8, 10, 13, etc. of file 'U-236.PreEq' shown above. For the other lines, the structure description in Table 8 applies.

Table 7: Structure of pre-equilibrium emission spectrum data file.

Column	Description
1	Entry number (Same as Column 1 in Table 6.)
2	Incident neutron energy, E_n, in units of MeV. (Same as Column 1 in Table 6.)
3	Number of entries in the pre-equilibrium neutron energy spectrum
4	Bin width of the pre-equilibrium energy spectrum in units of MeV. (Identical for all entries.)

The file 'U-236.PreEq' gives the emission spectrum out to its kinematic endpoint. The entry numbers and incident neutron energies are the same as those in the first two columns of the probability distribution, as described in Table 7. If the third entry in 'U-236.PreEq' is 0, there are no emission spectrum entries since there is zero probability for emission. If this entry is nonzero, the emission spectrum follows with a number of entries equal to the integer value of Column 3. Table 8 describes these entries.

Table 8: Description of pre-equilibrium emission spectrum for each incident neutron energy where the number of entries in the third column of Table 7 is not null.

Column	Description
1	Pre-equilibrium emission neutron energy.
2	Pre-equilibrium emission neutron spectrum, $d\sigma_n/dE$, pre-computed using Eq. (1), and normalized to unity.

The numbers of incident neutron energies and the widths of the incident neutron energy bins must match in the pre-equilibrium emission probability file (e.g. U-236.xs) and the pre-equilibrium emission spectra file (e.g. U-236.PreEq).

3.6 Data file 'Zdis.dat'

The data file 'Zdis.dat" is used to sample the fission fragment charge partition, see Fig. 3. It contains the standard deviation σ_Z in Eq. (3) for selected elements with $220 \leq A \leq 260$. An excerpt of the file is shown below:

```
  A  charge distribution
220                 0.38
221                 0.38
(...)
259                 0.47
260                 0.47
```

The overall structure of the data file ‘Zdis.dat’, aside from the first header line, is explained in Table 9.

Table 9: Structure of data file ‘Zdis.dat’.

Column	Description
A	Mass number
σ_Z	Width of charge distribution, standard deviation in Eq. (3)

3.7 Data files for fission fragment yields

In cases where either data on only a single incident energy is available or for some spontaneously fissioning isotopes, the fragment yields are sampled from the corresponding data file. The structure of these data files are described in Sec. 3.7.1. In other cases, a five-Gaussian fit to the fragment yields has been made. The structure of the data file containing the fit parameters, ‘gaussfit.dat’, is described in Sec. 3.7.2. (Both single data files and five-Gaussian fits are available for spontaneous fission. In general the data file is used for sampling in these cases.) When the full incident neutron energy dependence of the yields is required for energies up to $E_n = 20$ MeV, an energy dependence of the Gaussian fits has been developed, as discussed in Sec. 3.7.2.

3.7.1 Data files for single fission fragment yields $P(A_f)$

The data files for $P(A_f)$ for single energy or spontaneous fission fragment yields are used in the selection of the fission fragment mass numbers, see Fig. 3. The excerpt of the file ‘U234-Af-N.dat’ below shows $P(A_f)$ for thermal fission of the first compound nucleus in ‘react.dat’, $^{234}\mathrm{U}^*$.

```
P(Af) for 234U [Nishio]
 74  .2566E-01  .8890E-02
 75  .5132E-01  .1222E-01
(...)
159  .5132E-01  .5132E-02
160  .2566E-01  .2963E-02
  0  .0         .0
```

Aside from the first header line, the lines of the $P(A_f)$ files are described in Table 10. The fragment yields $P(A_f)$ do not need to be normalized, the code normalizes them automatically. The header line indicates the origin or author of the data, e.g. “Nishio”. The end of the data file is identified by a line where the fragment mass number A_f is 0.

Table 10: Structure of fission product yield probability data file $P(A_f)$.

Column	Description
1	Fragment mass number, A_f
2	Fragment yields $P(A_f)$
3	Uncertainty on $P(A_f)$

3.7.2 Data file 'gaussfit.dat'

When the file name for the fission fragment yield $P(A_f)$ is replaced by a hyphen '-' (e.g. ^{236}U* and ^{240}Pu* in file 'react.dat'), FREYA samples the fission products from a five-Gaussian fit to the fission product yield distributions. The fits are isotope and energy dependent, as is now described. See Ref. [20] for an example.

The mass yields $Y(A_f)$ of the fission fragments for a given neutron energy E_n are composed of three distinct Gaussian modes,

$$Y(A_f) \;=\; G_1(A_f) + G_2(A_f) + G_0(A_f)\ . \tag{17}$$

The first two terms represent asymmetric fission modes associated with the spherical shell closure at $N = 82$ and the deformed shell closure at $N = 88$, respectively. The last term is a symmetric mode. While this mode is small at low excitation energies, its importance increases with excitation energy.

The asymmetric modes are composed of two Gaussians,

$$G_i = \frac{C_n(i)}{\sqrt{2\pi}W_n(i)}\left[\exp\left(-\frac{(A_f - \overline{A} - D_n(i))^2}{2{W_n}^2(i)}\right) + \exp\left(-\frac{(A_f - \overline{A} + D_n(i))^2}{2{W_n}^2(i)}\right)\right]\ , \tag{18}$$

where $i = 1, 2$ while the symmetric mode is given by a single Gaussian

$$G_0 = \frac{C_n(0)}{\sqrt{2\pi}W_n(0)}\exp\left(-\frac{(A_f - \overline{A})^2}{2{W_n}^2(0)}\right)\ , \tag{19}$$

with $\overline{A} = A_0/2$.

The values of $D_n(i)$ are displacements that are anchored above the symmetry point by the spherical and deformed shell closures. Because these occur at specific neutron numbers, $D_n(i)$ are energy independent. The values of $D_n(i)$ are smaller for ^{240}Pu* than ^{236}U* due to the larger value of A_0 for Pu.

The widths of the asymmetric Gaussians are assumed to be energy dependent and are expended to second order in neutron energy,

$$W_n(i) = W_n(i,0) + W_n(i,1)E_n + W_n(i,2)E_n^2\ . \tag{20}$$

The width of the symmetric Gaussian is assumed to be energy independent.

The energy dependence of the normalization coefficients $C_n(1)$ and $C_n(2)$ is given as

$$C_n(i) = C_n(i,0)\,(1 + \exp[(E_n - C_n(i,1))/C_n(i,2)])^{-1}\ . \tag{21}$$

Spontaneous fission has no energy dependence. Therefore, $W_n(i,1) = W_n(i,2) = 0$ while $C_n(i,1) \equiv 1$ and $C_n(i,2) \equiv 0.05$ so that $C_n(i) \sim C_n(i,0)$ in Eq. (21). Since each event leads to two fragments, the yields are normalized so that $\sum_{A_f} Y(A_f) = 2$. Thus,

$$2C_n(1) + 2C_n(2) + C_n(0) = 2\ , \tag{22}$$

apart from a negligible correction because A_f is discrete and bounded from both below and above. Therefore, $C_n(0)$ is determined from Eq. (22) at each value of E_n.

Finally, we note that, above the threshold for pre-fission neutron evaporation, the yields include contributions from first-chance fission and higher. For more information, see Ref. [20].

The list of parameters used for the Gaussian fits are in the file ‘gaussfit.dat’, reproduced below. The structure of ‘gaussfit.dat’ is described in Table 11. Note that, at this time, there are no parameters included for $^{233}\mathrm{U}+n$ because the yields are directly sampled from the data.

```
Weight of each fission mode
 Z   A   Cn(1,0)  Cn(1,1)  Cn(1,2)
         Cn(2,0)  Cn(2,1)  Cn(2,2)
           Dn(1)    Dn(2)
         Wn(0,0)  Wn(0,1)  Wn(0,2)
         Wn(1,0)  Wn(1,1)  Wn(1,2)
         Wn(2,0)  Wn(2,1)  Wn(2,2)
92 236   0.73426    9.169   1.1887
         0.26570    9.169   1.1887
          23.05    16.54
           12.0      0.0      0.0
          5.077  0.09376    0.034
          2.978   0.1106    0.008
92 238  0.494508     1.0     0.05
        0.505472     1.0     0.05
           25.81    18.22
           4.832      0.0      0.0
           3.256      0.0      0.0
           3.313      0.0      0.0
94 240     0.757    10.14     1.15
           0.242    10.14     1.15
           20.05    14.54
            12.0      0.0      0.0
             5.6  0.09376    0.034
             2.5   0.1106    0.008
96 244   0.41987      1.0     0.05
        0.580064      1.0     0.05
           21.04    15.12
          0.7404      0.0      0.0
           5.542      0.0      0.0
           3.692      0.0      0.0
98 252  0.655921      1.0     0.05
        0.344077      1.0     0.05
           18.02    15.78
           14.01      0.0      0.0
           7.106      0.0      0.0
           5.546      0.0      0.0
```

3.8 Isotopic mass tables

The isotope mass tables used in the scission algorithm in Fig. 4 are described here. When available, the experimental measurements in ‘MassAudi.dat’ are used. Otherwise, the theoretical values in ‘MassMNMS.dat’ are used.

The structure of the two files is essentially identical. Both files have a five-line header, including two equations, identifying the components of the file. The first equation of the five-line header gives the ground

Table 11: Structure of fission product yield probability data file ‘gaussfit.dat’.

Section	Line#	Description
Header	1	Descriptive file header
	2-3	The first two variables on line 2 are the proton number Z and mass number A of the compound nucleus before fission. This isotope must be present in the file ‘react.dat’. The three parameters $C_n(i,j)$ on lines 2 and 3 describe the energy dependence of the normalization coefficients, see Eq. (21). While $C_n(i,0)$ is a number, $C_n(i,1)$ and $C_n(i,2)$ have units of MeV (see Eq. 21). The normalization of the symmetric mode is constructed from Eq. (22).
	4	The energy-independent values $D_n(i)$ are the centroid shifts of the asymmetric fission modes.
	5-7	The parameters $W_n(i,j)$ are the energy-dependent widths of the Gaussians describing the asymmetric fission modes, see Eq. (20). While $W_n(i,0)$ is dimensionless, $W_n(i,1)$ and $W_n(i,2)$ have units of MeV^{-1} and MeV^{-2} respectively.
Data	Each subsequent set of 6 lines corresponds to the five-Gaussian fit parameters for the indicated Z A combination.	
	8-13	Gaussian fit parameters for first Z A combination
	14-19	Gaussian fit parameters for second Z A combination
	20-35	(...)

state mass of the nucleus,

$$M(Z,A) = Au + D(Z,A) \ , \tag{23}$$

where Au is the mass number times the atomic mass unit, u, and $D(Z,A)$ is the mass defect, all in units of MeV. The numerical value for u is given in the third line of the header file, also in units of MeV. The second equation, on the fourth line of the header, defines the nuclear binding energy

$$B(Z,A) = ZD(H) + ND(n) - D(Z,A) \ . \tag{24}$$

The binding energy of the compound nucleus is the difference between the sum of the mass defects for all nucleons and the mass defect of the compound nucleus.

The structure of the data in the files ‘MassMNMS.dat’ and ‘MassAudi.dat’ following the header is explained in Table 12. To indicate the end of the data files, the last line of the file has five zero values.

Table 12: Structure of data files ‘MassMNMS.dat’ and ‘MassAudi.dat’.

Column	Description
Z	Proton number
A	Mass number
M	Ground state isotopic mass in units of MeV
D	Mass defect in units of MeV
B	Binding energy in units of MeV

3.8.1 Theoretical isotopic mass tables in data file ‘MassMNMS.dat’

The theoretical isotopic masses, mass defects and binding energies are taken from Möller *et al.* [28]. The excerpt from the file ‘MassMNMS.dat’ shows some of the theoretical masses. Note that the lowest tabulated theoretical mass is for ^{16}O, $Z = 8$, $A = 16$.

```
 * Theoretical masses from MNMS 1995:
 * M(Z,A) [MeV] = A*u + D(Z,A) where u is
  931.493835 MeV
 * B(Z,A) = Z*D(H) + N*D(n) - D(Z,A)
 *  Z    A      M(MeV)       D(MeV)       B(MeV)
    8   16   14899.0615     -4.84000    127.72232
    8   17   15835.2256     -0.17000    131.12367
(...)
  120  299  278729.9688    213.32001   2106.12451
  120  300  279662.2812    214.14000   2113.37573
    0    0       0.0000      0.00000      0.00000
    0    0       0.0000      0.00000      0.00000
```

3.8.2 Experimental isotopic mass tables in data file ‘MassAudi.dat’

This table contains the experimentally measured isotopic masses, mass defects and binding energies published by Audi and Wapstra [27]. Whenever available, the experimental values will override the theoretical ones in Sec. 3.8.1. An excerpt of this file is shown below. Note that the neutron ($Z = 0$, $A = 1$) and proton ($Z = 1$, $A = 1$) are the first two entries in this table.

```
 * Audi & Wapstra NPA595 (1995) 409:
 * M(Z,A) [MeV] = A*u + D(Z,A) where u is
  931.493835 MeV
 * B(Z,A) = Z*D(H) + N*D(n) - D(Z,A)
 *  Z    A      M(MeV)       D(MeV)       B(MeV)
    0    1     939.5652      8.07135      0.00000
    1    1     938.7828      7.28894      0.00000
(...)
  117  292  272189.5312    193.33130   2071.96094
  118  293  273127.6562    199.96246   2072.61865
    0    0       0.0000      0.00000      0.00000
```

3.9 Total kinetic energy data file

FREYA needs to read in the total kinetic energy TKE data files, given as a function of the heavy fragment mass A_H. One or more TKE files per isotope are allowed. The number of TKE files for a given isotope is specified in the field ‘# TKE files’ in ‘react.dat’. When multiple TKE files are entered, the code calculates the average of the experimental TKE values for each A_H, counting each data point equally. (Therefore A_H and $A_L = A_0 - A_H$ count equally). An entry of 0 in the field ‘# TKE files’ is invalid because there is no default behavior for the total kinetic energy data. If the TKE data file is missing or invalid, FREYA will generate an error message.

The total kinetic energy data files are used in the algorithm for the selection of fission fragment energies, see Fig. 5 and Eq. 5. An example of a TKE file is shown below for neutron-induced fission of ^{235}U, for the compound nucleus ^{236}U*:

```
Ktot(AH) for 236U thermal [Nishio]
118 156.5537
119 158.1543
```

```
(...)
157 149.8098
158 148.0319
  0   0.0
```

The first line indicates the compound nucleus and the origin or author of the data, e.g. "Nishio". The last line is identified by at least two numbers, the first one of which must be 0.

Table 13: Structure of total kinetic energy data file.

Column	description
A_H	Mass number of heavy fission fragment
TKE	Total kinetic energy in units of MeV, Eq. 5

3.10 Data file 'inputparameters.dat'

The file 'inputparameters.dat' contains the parameters used to calculate the total excitation energy of the fission fragments and partition the excitation energy between the two fragments (see Eqs. (5)-(10) in Sec. 2). This file enters into the algorithm for the selection of fission fragment energies (Fig. 5) and the algorithm for the neutron evaporation and photon emission (Fig. 6).

This file also includes the parameter dTKE which shifts the TKE globally to get agreement with the measured average neutron multiplicity, $\overline{\nu}$. If a hyphen '-' is given in the field 'dTKE file name' or if this field is empty, FREYA uses the energy-independent value specified in the next field 'dTKE'. An entry in the column 'dTKE file name' is only expected for neutron-induced fission at multiple energies, for E_n up to 20 MeV.

If isotopes in file 'react.dat' have no entries in file 'inputparameters.dat', the unspecified fields 'alevel0', 'xeps' and 'dTKE' take the species-independent default values of 10.0724 MeV, 1.23389 and 1.53729 MeV.

```
  Z   A  reaction   alevel0  xeps   dTKE file name   dTKE
92  234   '(n,f)'   10.0724  1.1    -                 0.
92  236   '(n,f)'   10.0724  1.2    U-236.dTKE        0.39
92  238   'sf'      10.0724  1.2    -                 0.2
94  240   '(n,f)'   10.0724  1.1    Pu240.dTKE        1.1
94  240   'sf'      10.0724  1.2    -                -0.5
96  244   'sf'      10.0724  1.2    -                -2.375
98  252   'sf'      10.0724  1.3    -                 2.75
```

3.11 Data file 'alevel.dat'

The file 'alevel.dat' contains the values δW and Δ in Eq. (8) which are used to include shell corrections and pairing effects in the level-density parameter a_f (see function 'alevel' in FORTRAN code). The data in this file enters whenever the level density is used, not only when dividing the statistical excitation between the two fragments but also when sampling the energy of an emitted neutron or photon. The flow charts indicate that 'alevel.dat' is used to determine fission fragment thermal fluctuations (see Fig. 5) as well as for neutron evaporation and photon emission (Fig. 6). An excerpt of file 'alevel.dat' is shown below.

```
   Z    A   deltaW     Delta   [notes at end of file]
  19   58   3.45        0.00
  19   59   2.51        1.21
```

Table 14: Structure of data file ‘inputparameters.dat’.

Column	Description
Z	Proton number
A	Mass number of compound nucleus
reaction	Reaction type, either neutron-induced fission ‘(n,f)’ or spontaneous fission ‘sf’
alevel0	Asymptotic level density parameter e_0 (see Eq. (8) and Ref. [20])
xeps	Parameter for asymmetric distribution of excitation energy between the light and heavy fragments (see Eq. (9))
dTKE file name	Name of file containing $d\mathrm{TKE}(E_n)$ (see Eq. (5)).
dTKE	Parameter value dTKE for spontaneous or thermal fission in units of MeV (see Eq. (5))

```
(...)
   94  239  -2.12      1.00
   94  240  -1.95      1.66
    0    0   0.00000   0.00000
    Z    A   deltaW    Delta
----------------------------------------------
01dec09 (JR):: Entries added for Z=19 & 94
which are needed for 239Pu(n,f); the
shell energy deltaW was taken as Emic
from MNMS, while the pairing energy
Delta was extrapolated from nearby
values already in the table.
* Remember to increase the parameter ’max’
in alevel when augmenting the table.
----------------------------------------------
```

Aside from the first header line and the last comment lines, the overall structure of the data in this file is described in Table 15. The data file ‘alevel.dat’ must contain values of Z, A, δW, Δ in order of uninterrupted, increasing Z. The values for each Z in order of uninterrupted, increasing A are nested within the increasing Z values. Of the 7076 current entries, 7050 entries are from Koura [30]. An additional 18 entries for $Z = 19$ and $Z = 94$ were added for ^{239}Pu(n,f). For these extra entries, the shell energy δW was taken to be E_{mic}, the microscopic (pairing and shell) part of the nuclear potential energy of deformation, from Ref. [28], while the pairing energy Δ was extrapolated from nearby values already in the table. The line following the last data line is identified by at least four numbers, the first one of which must be $Z = 0$.

Table 15: Description of structure for data file ‘alevel.dat’.

column	description
Z	Proton number
A	Mass number
deltaW	Shell correction δW (see Eq. (8))
Delta	Pairing energy Δ (see Eq. (8))

FREYA assumes a maximum of 7076 entries for this file, so if additional entries are added, the maximum parameter in FREYA needs to be augmented as well.

4 Integration of FREYA into LLNL Fission Library

The Monte-Carlo codes MCNPX, MCNP6, and Geant4.10 already had an interface to the LLNL Fission Library for sampling fission events. To more efficiently incorporate FREYA into these Monte-Carlo codes, FREYA was integrated into the LLNL Fission Library version 1.9. This was convenient because no modifications of these Monte-Carlo codes were necessary after the updated LLNL Fission Library, including FREYA, was substituted into their source code trees. Fig. 7 shows schematically how the Monte-Carlo codes communicate with the LLNL Fission Library and FREYA. The full documentation of the LLNL Fission Library [14] is available at the following URL: http://nuclear.llnl.gov/simulation/ in the documentation of the section titled "Fission". As far as maintenance and distribution is concerned, our plan is to update the LLNL Fission Library with newer versions of FREYA whenever available. (Maintenance of the LLNL Fission Library is also foreseen.)

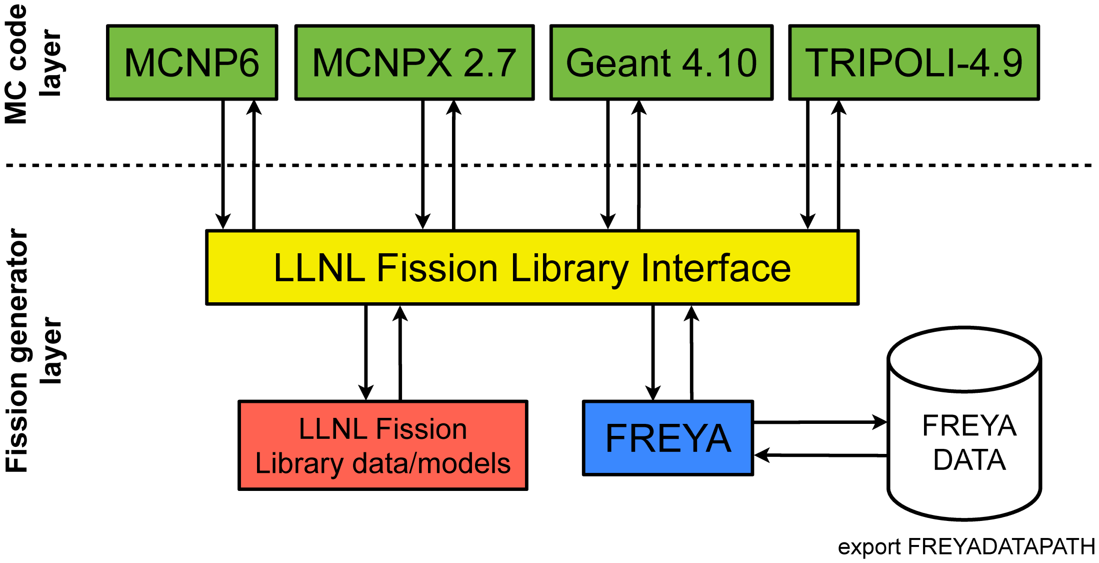


Figure 7: Schematic of Monte Carlo codes communicating with FREYA through the LLNL Fission Library interface.

As stated in Sec. 3, FREYA is currently limited to 4 spontaneous fission isotopes and 3 neutron-induced isotopes[3,4]. FREYA selects outgoing projectiles from spontaneous fission and neutron-induced fission for incident neutron energies below 20 MeV[5].

For each spontaneous or neutron-induced fission event, the code checks whether the sampled isotope is available in FREYA. If the isotope is included, FREYA is called to sample multiplicity, energy, and direction of the fission neutrons, all of which are passed back to the LLNL Fission Library and eventually to the host

[3] Photofission is planned for the near future.

[4] If a reaction (spontaneous fission or neutron-induced fission) for an isotope is not specified in the master data file 'react.data', an error message will be generated and the LLNL Fission Library will temporarily revert to correlation option 0 (see Sec. A.1.7) for this reaction. Error messages can be retrieved by the user, see Secs. A.1 and A.2.

[5] If the energy of an incident neutron is greater than 20 MeV or if one of the error conditions described in Sec. 3 occurs, the LLNL Fission Library will temporarily revert to correlation option 0 (see Sec. A.1.7).

Monte-Carlo code for transport. All fission events of isotopes not in FREYA are handled by the default LLNL Fission Library settings in the usual way. When in use, FREYA predicts a host of correlations between outgoing fission products: correlations in neutron multiplicity, energy and angles, and energy sharing between neutrons and photons.

It is not necessary to run the aforementioned Monte Carlo codes to access the LLNL Fission Library. Modulo a linking step, a user can call the latter directly through its C and C++ interfaces with standalone applications. This will be shown in the codes "angular_correlation.c", "angular_correlation.cpp", "nu_distribution.cpp" and "fission_neutron_spectra.cpp" described below. Moreover, it is also possible to call FREYA's FORTRAN routines directly. All FREYA routines have C bindings. This is the path that was chosen for the code "ff_yield.cpp", see these results in Fig. 11. Because this is not the recommended access pattern, details on how to access the FREYA routines directly will not be given here.

We verified [31] that FREYA yields the correct average neutron induced fission spectrum within MCNP by calculating the criticality parameter k_{eff} for the critical assemblies Godiva and Jezebel [32]. The k_{eff} results using FREYA were 0.9994±0.0009 (Jezebel) and 1.0003± 0.0008 (Godiva), in good agreement with the default MCNP values. In a suite of regression tests, values of $\bar{\nu}$ (average numbers of fission neutrons per fission) are regularly compared to experimental data to make sure code regressions are not introduced. In Ref. [33], preliminary results show that FREYA simulations are in agreement with experimental data. FREYA results are also successfully compared to previously published neutron-neutron angular correlation data in Refs. [34, 35].

5 Input/Output

5.1 Random number generator

By default, the LLNL Fission Library uses drand48() from the BSD library as a random number generator. The user can select a different random number generator via a call to the functions in Sec. A.1.10 or their C counterparts. To be able to reproduce identical sequences of random numbers, the aforementioned Monte Carlo codes pass their own random number generator to the LLNL Fission Library.

FREYA will use the same random number generator as the LLNL Fission Library if it is accessed through the latter. Otherwise, it will use ran1() from Numerical Recipes [36]. Properly seeding the random number generator is left to the user or to the host Monte Carlo code.

5.2 Input parameters

A fission event will be generated based on multiple input parameters:

i) ZA of the spontaneously fissioning nucleus, or of the nucleus before absorbing the incoming neutron for the case of neutron-induced fission;

ii) $\bar{\nu}$, the average number of neutrons emitted per fission;

iii) E_n, the incident neutron energy;

iv) fission type (spontaneous fission, neutron-induced fission).

These input parameters are not consistently used in all cases. For instance, FREYA never uses $\bar{\nu}$, but the LLNL Fission Library uses $\bar{\nu}$ when FREYA cannot handle the requested nucleus; for spontaneous fissions, $\bar{\nu}$ and E_n are never used. The Monte Carlo host codes mentioned above provide the input parameters to the LLNL Fission Library.

FREYA was designed to handle nuclei in motion. However, the LLNL Fission Library interface only allows nuclei at rest.

5.3 Output parameters

Based on the input parameters provided above, FREYA creates a number of secondary particles (neutrons and photons). The following quantities are provided for each:

i) momentum vector;

ii) kinetic energy;

iii) particle identification (0 for photon, 1 for neutron).

It also produces the following observables for the two fission fragments:

i) excited energy;

ii) momentum vector;

iii) kinetic energy;

iv) charge;

v) mass number.

Unfortunately, this last set of observables is currently not accessible via the LLNL Fission Library interface. To access these observables, one needs to call the FREYA subroutines directly, see sample code “ff_yield.cpp” for this capability. One should point out that FREYA does not return information pertaining specifically to pre-equilibrium neutron emission or pre-fission neutron evaporation. It is thus impossible to know which process produced each of the secondaries.

6 Typical FREYA run sequence

This section illustrates how to use FREYA within the LLNL Fission Library. Sample C and C++ codes are distributed with the LLNL Fission Library/FREYA package[6], and call the fission library via the application programmable interfaces (APIs). It is also possible to directly issue FORTRAN calls to FREYA subroutines and functions in the fission library from a FORTRAN program.

Two codes (`angular_correlation.cpp` and `angular_correlation.c`) are presented in Appendices B and C. They both create histograms exhibiting the angular correlation between two neutrons emitted during a fission event. The first uses the C++ API of Appendix A.1, while the second uses the C API of Appendix A.2. The third code (`nu_dist.cpp`) computes the multiplicity distribution of neutrons emitted per fission. The fourth (`fission_neutron_spectra.cpp`) computes the prompt fission neutron spectra for different neutron multiplicities. The last (`ff_yield.cpp`) bypasses the interfaces of Appendices A.1 and A.2 to call the FREYA functions directly. It computes the fission fragment yield and the total kinetic energy of the fission fragments as a function of the fission fragment mass.

6.1 Installation of the library

The LLNL Fission Library/FREYA package is available at the following URL: http://nuclear.llnl.gov/simulation/. The FREYA part of the package is written in Fortran 90 and has been tested with both gfortran and the Intel Fortran compiler. After obtaining the FREYA distribution, the user should unpack the library using the command

[6] See `Readme.cmake` in subdirectory `fission_v1.9/sample_codes`.

```
tar -xzvf fission_v1.9.tar.gz
```

The LLNL Fission library/FREYA package is built as a dynamic library using cmake (see Readme's in fission_v1.9/ and fission_v1.9/src)

```
cd fission_v1.9
mkdir build; cd build
cmake ../src; make install
```

As a result, a dynamic library is created in directory fission_v1.9/lib, and can be linked in with test codes or host radiation transport codes.

6.2 Building the test cases

To build the test cases, the user should use cmake. For two neutron correlations using the C++ interface, the user executes the following commands:

```
cd fission_v1.9/sample_codes/angular_correlation_cpp
mkdir build; cd build
cmake ../src; make
```

6.3 Running the simulations

To run, the environment variable FREYADATAPATH must point to the directory containing the data files used by FREYA. In the bash shell, this can be done with the following statement:

```
export FREYADATAPATH=/path/to/freya/data/directory/
```

If unspecified, the LLNL Fission Library will look into the directory './data' to find the data for FREYA[7]. The environment variable LD_LIBRARY_PATH[8] needs to point to the location of the shared library libFission. Once these 2 variables are set, the simulation can be run

```
./angular_correlation
```

GNUPLOT can be used to visualize the results:

```
cd ../plot
gnuplot angular_correlation.gnu -
```

The first two example codes (angular_correlation.c and angular_correlation.cpp) produce the graph in Fig. 8, the angular correlations between two neutrons emitted in ^{239}Pu(n,f) with 2 MeV incident neutrons. The third (nu_dist.cpp) and fourth (fission_neutron_spectra.cpp) codes produce Fig. 9, the neutron multiplicity distribution for ^{235}U(n,f) with $E_n = 1$ MeV, and Fig. 10, the prompt fission neutron spectrum as a

[7] If a valid location for the data is not found, an error message will be generated and the LLNL Fission Library will run under correlation option 0 (see Sec. A.1.7) with FREYA turned off.

[8] DYLD_LIBRARY_PATH on MacOS X

function of outgoing energies for specific neutron multiplicities. The last (ff_yield.cpp) computes the fission fragment yield as a function of fragment mass, see Fig. 11, and the total kinetic energy of the fission fragments as a function of the heavy fragment mass, see Fig. 12.

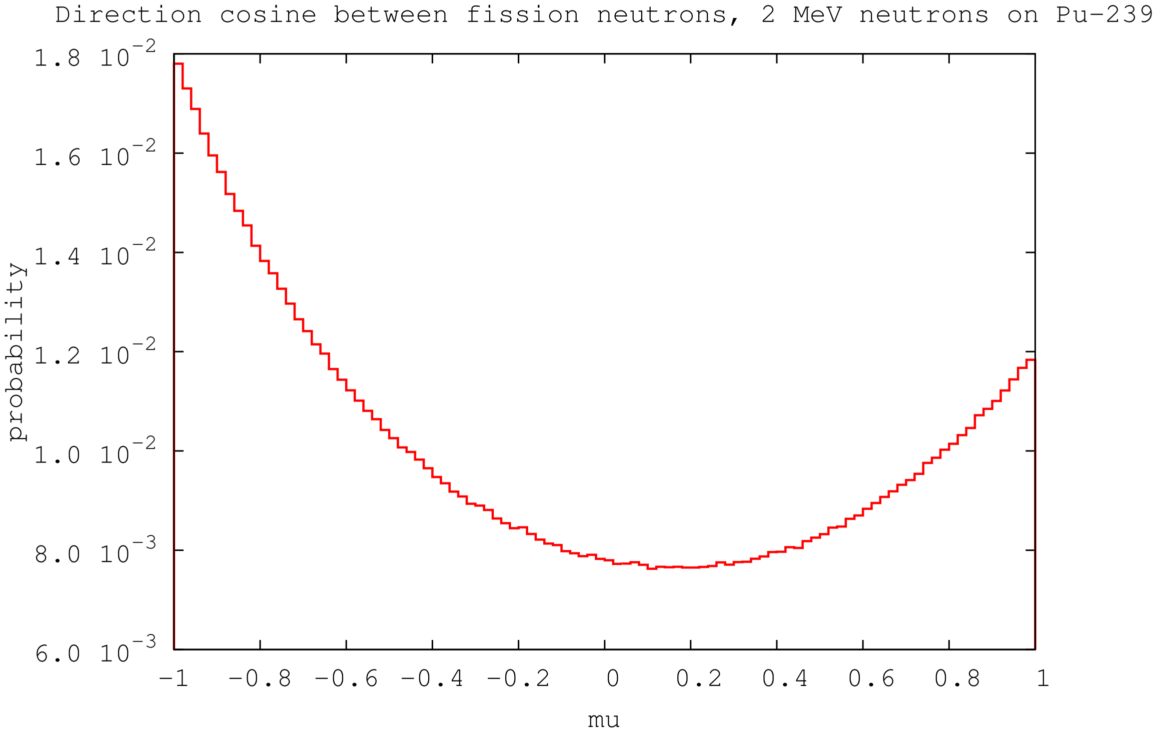


Figure 8: Output of the sample codes angular_correlation.c and angular_correlation.cpp showing the two-neutron correlation as a function of the angle between the neutrons emitted from fission of ^{239}Pu induced by 2 MeV neutrons.

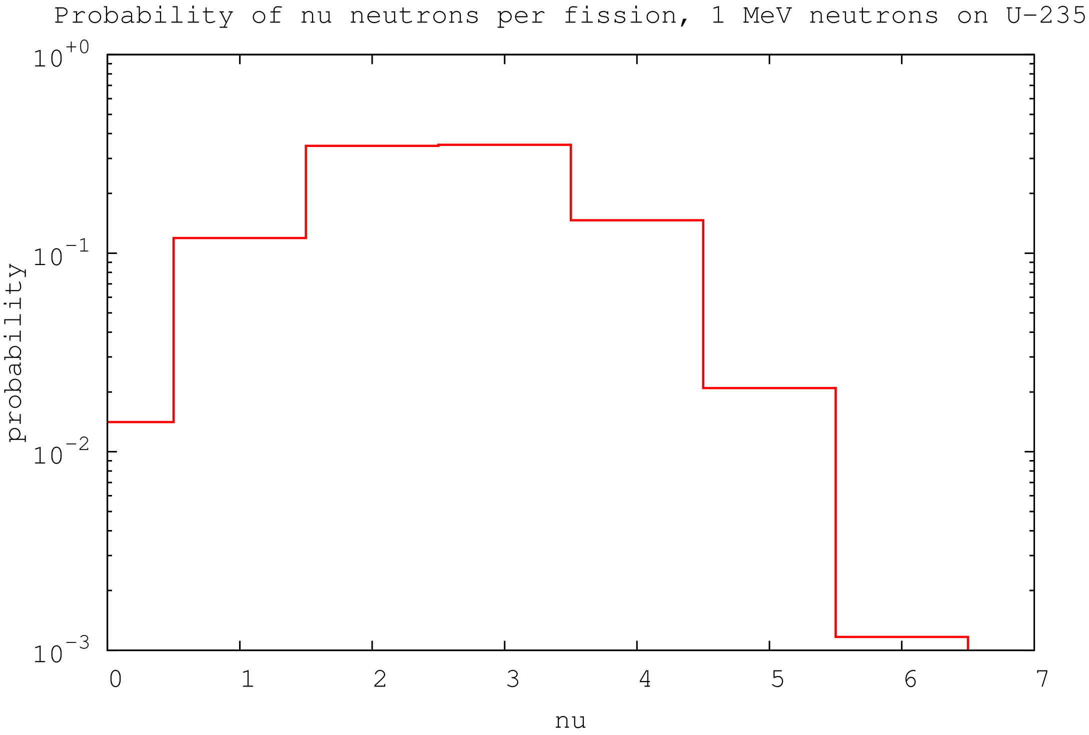


Figure 9: Output from the sample code `nu_distribution.cpp` showing the neutron multiplicity distribution obtained from fission of ^{235}U induced by 1 MeV neutrons.

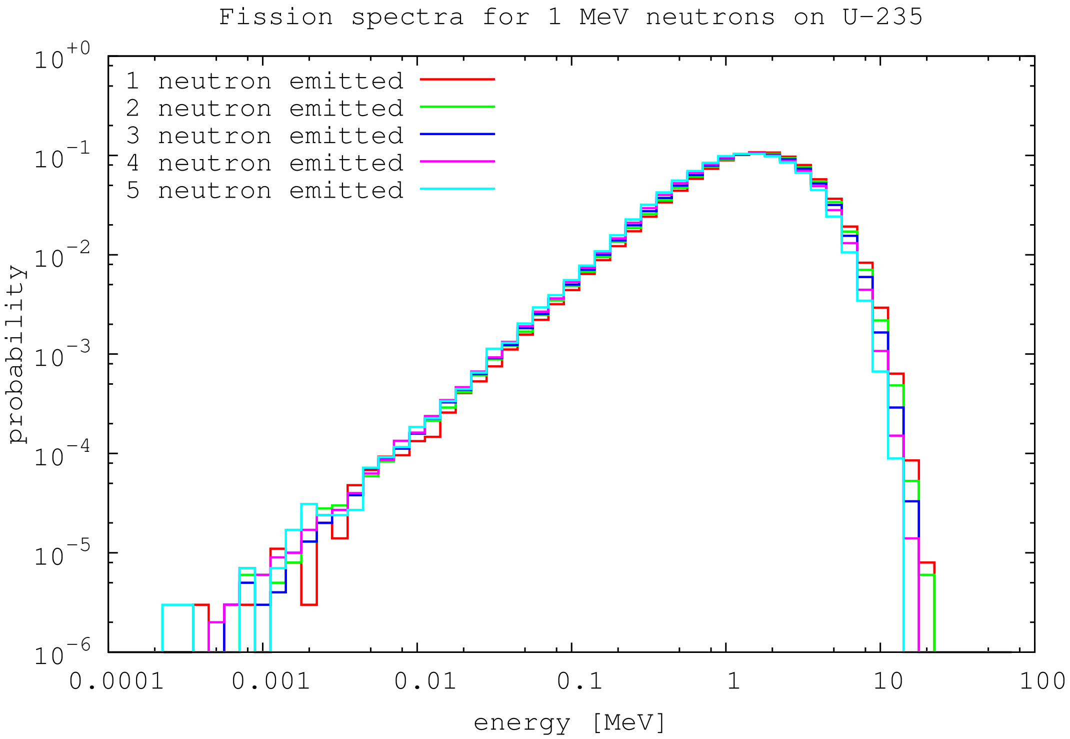


Figure 10: Output from the sample code `fission_neutron_spectra.cpp` showing the prompt fission neutron spectra for different neutron multiplicities from fission of ^{235}U induced by 1 MeV neutrons. The spectra for neutron multiplicities of 1 through 5 are given.

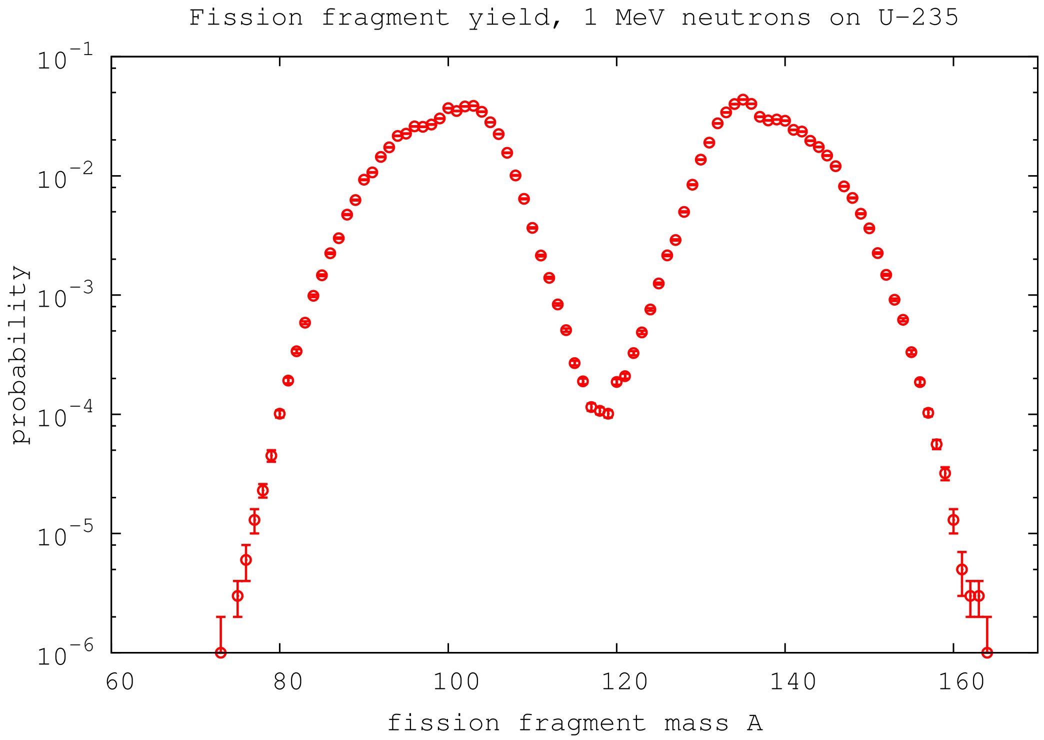


Figure 11: Output from the sample code `ff_yield.cpp`. The fission fragment yield as a function of fragment mass for fission of ^{235}U induced by 1 MeV neutrons is shown.

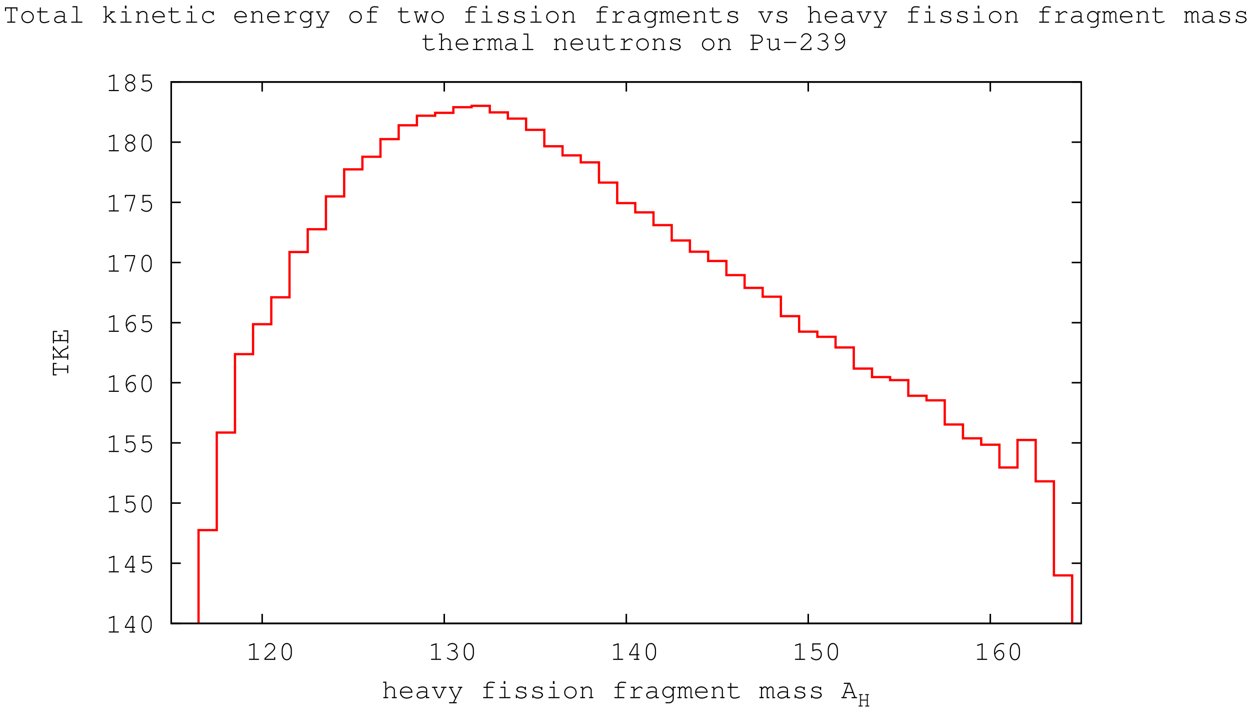


Figure 12: Output from the sample code `ff_yield.cpp`. The average total kinetic energy of fission fragments is shown as a function of heavy fragment mass for fission of ^{239}Pu induced by thermal neutrons.

7 Conclusion

This manual describes the event-by-event fission generator FREYA and its integration into the LLNL Fission Library. The upgraded LLNL Fission Library was used within MCNPX2.7.0 to run Monte Carlo neutron transport simulations and to verify that results conformed to expectations for criticality benchmarks.

This new FREYA capability enables the simulation of correlations that are not predicted by conventional neutron Monte Carlo codes. For instance, the results in Refs. [33–35] show that measured angular correlations of fission neutrons can be qualitatively reproduced with FREYA.

Several improvements of FREYA are planned, such as a more refined treatment of fission photons, the addition of more isotopes, and photofission. Currently, only prompt fission secondaries are emitted. While delayed emission could be added in the future, there are no immediate plans to do so. We are working with the stochastic neutron transport community to make the LLNL Fission Library/FREYA package publicly available to the wider community.

8 Acknowledgments

This work performed under the auspices of the U.S. Department of Energy by Lawrence Livermore National Laboratory under Contract DE-AC52-07NA27344 and by Lawrence Berkeley National Laboratory under Contract DE-AC02-05CH11231.

J. Verbeke wishes to thank the stochastic team (LTSD) of the Service d'Etudes de Réacteurs et de Mathématiques Appliquées (SERMA) in the Commissariat à l'Energie Atomique et aux Energies Alternatives at Saclay, France for hosting him while this paper was being written and while FREYA was being integrated into TRIPOLI-4.9. Odile Petit made invaluable contributions to that integration. He is also grateful to Mike James and Gregg McKinney from the MCNPX development team at Los Alamos National Laboratory for helping him integrate the LLNL Fission Library/FREYA package into MCNPX2.7.0 and MCNP6.

A Application programmable interfaces (APIs)

FREYA is accessed via the API to the LLNL Fission Library. There are currently two different APIs in C and in C++. A description of the full interface can be found in Ref. [14]. The parts of the APIs that are relevant to FREYA will be presented here.

A.1 C++ API

The C++ interface to the LLNL fission library consists of a number of C++ functions. The functions that are relevant to FREYA are described below.

A.1.1 fissionEvent::fissionEvent(int isotope, double time, double nubar, double eng, int fissiontype)

The constructor of class **fissionEvent**. It is called to generate a fission event. Multiple neutrons and photons are generated and stored in a stack along with their energies, directions and emission times. The arguments of this function are:

isotope:	entered in the form *ZA* (e.g. 94239 for ^{239}Pu).
time:	time of the spontaneous fission.
nubar:	user-specified average number of neutrons emitted per fission (e.g. as tabulated in the cross section libraries used by the particle transport code).
eng:	energy of the neutron that induces fission.
fissiontype:	type of fission: 0) spontaneous fission, 1) neutron-induced fission, 2) photofission.

Either the average number $\overline{\nu}$ of neutrons emitted per fission or the energy **eng** of the fission inducing neutron will be used to determine the number of neutrons sampled, see the function **setNudistOption()** below. The number of photons sampled only depends on $\overline{\nu}$.

A.1.2 fissionEvent::∼fissionEvent()

Destructor.

A.1.3 int fissionEvent::getNeutronNu()
int fissionEvent::getPhotonNu()

These functions return the number of fission neutrons and photons emitted in the fission reaction, or -1 if no number could be sampled in the fission library due to lack of data. The reader is referred to the physics reference manual [14] to find the list of isotopes for which sampling will return positive numbers.

A.1.4 double fissionEvent::getNeutronEnergy(int index),
double fissionEvent::getPhotonEnergy(int index),
double fissionEvent::getNeutronVelocity(int index),
double fissionEvent::getPhotonVelocity(int index)

These functions return the energies and velocity magnitudes of the neutrons and photons.

A.1.5 double fissionEvent::getNeutronDircosu(int index), double fissionEvent::getNeutronDircosv(int index), double fissionEvent::getNeutronDircosw(int index), double fissionEvent::getPhotonDircosu(int index), double fissionEvent::getPhotonDircosv(int index), double fissionEvent::getPhotonDircosw(int index)

These two function families return the direction cosines of the fission neutron and photon velocity vectors along the x, y and z axes.

A.1.6 double fissionEvent::getNeutronAge(int index) double fissionEvent::getPhotonAge(int index)

These functions returns the age of the fission neutrons and photons, or -1 if the index is out of range. Currently, delayed fission neutrons and photons are not implemented. Thus all fission products are the result of prompt emission.

A.1.7 static void fissionEvent::setCorrelationOption(int correlation)

This function is called to set the type of neutron-photon correlation. The argument **correlation** is set to

0 (default) for no correlation between neutrons and photons.

1 if the total fission neutron energy and total fission photon energy are sampled from normal distributions with means given in Beck *et al.* [37]. There is no correlation between the number of neutrons and the number of photons.

2 if the total fission neutron energy and total fission photon energy are sampled from normal distributions with means given in Vogt [38]. There is no correlation between the number of neutrons and the number of photons.

3 for the `FREYA` mode. The neutrons and photons are correlated in number and energy. If the isotope is included in `FREYA`, options set by **setnudist_()** and **setcf252_()** are ignored. Otherwise, it reverts back to correlation option 0.

A.1.8 static void fissionEvent::setNudistOption(int nudist)

This selects the data to be sampled for the neutron number distributions in the case of neutron-induced fission. If there are no data available, then the Terrell approximation is used for all cases. The argument **nudist** takes 4 values.

0 Use the fit to the Zucker and Holden tabulated $P(\nu)$ distributions as a function of energy for ^{235}U, ^{238}U and ^{239}Pu [39].

1 Use fits to the Zucker and Holden tabulated $P(\nu)$ distribution as a function of energy for ^{238}U and ^{239}Pu [39], and a fit to the Zucker and Holden data [39] as well as the Gwin, Spencer and Ingle data (at thermal energies) [40] as a function of energy for ^{235}U.

2 Use the fit to the Zucker and Holden tabulated $P(\nu)$ distributions as a function of $\overline{\nu}$ [39]. The ^{238}U fit is used for the ^{232}U, ^{234}U, ^{236}U and ^{238}U isotopes, the ^{235}U fit for ^{233}U and ^{235}U, the ^{239}Pu fit for ^{239}Pu and ^{241}Pu.

3 (default) Use the discrete Zucker and Holden tabulated $P(\nu)$ distributions and corresponding values of $\overline{\nu}$ [39]. Sampling based on the incident neutron $\overline{\nu}$. The ^{238}U data tables are used for the ^{232}U, ^{234}U, ^{236}U and ^{238}U isotopes, the ^{235}U data for ^{233}U and ^{235}U, the ^{239}Pu data for ^{239}Pu and ^{241}Pu.

For option 3 in **setCorrelationOption()**, `FREYA` computes the number of sampled fission neutrons independently and ignores this setting.

A.1.9 static void fissionEvent::setCf252Option(int ndist, int neng)

This function is specific to the spontaneous fission of ^{252}Cf. It selects the data to be sampled for the neutron number and energy distributions and takes the following arguments:

ndist: Sample the number of neutrons

0 (default)	from the tabulated data measured by Spencer [41].
1	from the Boldeman data [42].

neng: Sample the spontaneous fission neutron energy

0 (default)	from the Mannhart-corrected Maxwellian spectrum [43].
1	from the Madland-Nix model spectrum [44].
2	from the Watt spectrum [45] fit attributed to Fröhner [46].

For option 3 in **setCorrelationOption()**, `FREYA` computes the number and energies of sampled fission neutrons independently and ignores this setting.

A.1.10 static void fissionEvent::setRNGf(float (*funcptr) (void)), static void fissionEvent::setRNGd(double (*funcptr) (void))

This function sets the random number generator to the user-defined one specified in the argument. If either **setRNGf()** or **setRNGd()** are not specified, the default system call **srand48** is used. The arguments are random number generator functions that returns variables of type float and double respectively. The C++ language imposes that the function pointer in argument be either a global function or a static function of another class.

A.1.11 void fissionEvent::getFREYAerrors(int* length, char* errors)

This function returns potential error messages generated by `FREYA`. The arguments of this function are

Input

errors:	pointer to an allocated array of characters
length:	size of allocated array of characters passed in

Output

errors:	error message (if any)
length:	size of error message (=1 if no errors, >1 if errors)

A.2 C API

The C interface to the LLNL fission library consists of 23 C functions. The functions relevant to `FREYA` are described below.

A.2.1 void genspfissevt_(int *isotope, double *time)

This function is called to trigger a spontaneous fission event. Multiple neutrons and photons are generated and stored in a stack along with their energies, directions and emission times. The arguments of this function are

isotope: entered in the form *ZA* (e.g. 94239 for ^{239}Pu)
time: time of the spontaneous fission event.

The generated neutrons and photons, along with their properties, will be lost upon the next call to **genspfissevt_()** or **genfissevt_()**. Therefore, they must be retrieved by the caller before a subsequent call to one of these functions, using the appropriate functions described below.

A.2.2 void genfissevt_(int *isotope, double *time, double *nubar, double *eng)

This function is called to trigger a neutron-induced fission event. In addition to the two arguments above for **genspfissevt_()**, the fission-inducing neutron is characterized by:

nubar: user-specified average number of neutrons emitted per fission (e.g. as tabulated in the cross section libraries used by the particle transport code)
eng: energy of the neutron that induces fission.

Either the average number $\overline{\nu}$ of neutrons emitted per fission or the energy **eng** of the fission inducing neutron will be used to determine the number of neutrons sampled, see the function **setnudist_()** below. The number of photons sampled only depends on $\overline{\nu}$. Similar to **genspfissevt_()**, the generated neutrons and photons are lost upon subsequent calls to **genspfissevt_()** or **genfissevt_()**.

A.2.3 int getnnu_()
int getpnu_()

These functions are the counterparts of those in Sec. A.1.3.

A.2.4 double getneng_(int *index), double getpeng_(int *index)
double getnvel_(int *index), double getpvel_(int *index)

These functions are described in Sec. A.1.4.

A.2.5 double getneng_(int *index), double getpeng_(int *index)
double getnvel_(int *index), double getpvel_(int *index)

These functions are identical to those in Sec. A.1.4.

A.2.6 double getndircosu_(int *index), double getndircosv_(int *index), double getndircosw_(int *index)
double getpdircosu_(int *index), double getpdircosv_(int *index), double getpdircosw_(int *index)

These functions are explained in Sec. A.1.5.

A.2.7 double getnage_(int *index)
double getpage_(int *index)

These functions are the counterparts of those in Sec. A.1.6.

A.2.8 void setcorrel_(int *correlation)

This function is explained in Sec. A.1.7.

A.2.9 void setnudist_(int *nudist)

This function has its counterpart in Sec. A.1.8.

A.2.10 void setcf252_(int *ndist, int *neng)

This function is described in Sec. A.1.9.

A.2.11 void setrngf_(float (*funcptr) (void)), void setrngd_(double (*funcptr) (void))

These functions have their counterpart in Sec. A.1.10.

A.2.12 void getfreya_errors_(int* length, char* errors)

This function returns potential error messages generated by `FREYA`. Its arguments are described in Sec. A.1.11.

B Code using the C++ API

This first code "angular_correlation.cpp" uses the C++ interface described in Sec. A.1 to invoke the LLNL Fission Library.

```
#define iterations 3000000
#define nbins 100

#include <stdio.h>
#include "fissionEvent.h"

void init(void);
FILE* openfile(char* name);
void output(int* hist);

int main() {
   int isotope = 94239;
   double energy_MeV = 2.;
   double nubar = 3.163;
   double time = 0.;

   int maxerrorlength=10000;
   char errors[maxerrorlength];

   int hist[nbins];
   for (int i=0; i<nbins; i++) hist[i] = 0.;

   init();
   for (int i=0; i<iterations; i++) {
      fissionEvent* fe = new fissionEvent(isotope, time, nubar, energy_MeV, 1);
      int errorlength=maxerrorlength;
      fe->getFREYAerrors(&errorlength, &errors[0]);
      if (errorlength>1) {
         printf("%s\n",errors);
         exit(1);
      }
      int nneutrons = fe->getNeutronNu();
      for(int n1=0; n1<nneutrons; n1++) {
         double u1 = fe->getNeutronDircosu(n1), v1 = fe->getNeutronDircosv(n1), w1 = fe->getNeutronDircosw(n1);
         for(int n2=n1+1; n2<nneutrons; n2++) {
            double u2 = fe->getNeutronDircosu(n2), v2 = fe->getNeutronDircosv(n2), w2 = fe->getNeutronDircosw(n2);
            double scalar_product = u1*u2+v1*v2+w1*w2;

            int bin_index = (int) (nbins*(scalar_product+1)/2);
            hist[bin_index]++;
         }
      }
      delete fe;
   }
   output(hist);
}

void init(void) {
   unsigned short int s[3] = {1234, 5678, 9012};
   int i;
   seed48(s);
   fissionEvent::setCorrelationOption(3);
   return;
}

FILE* openfile(char* name) {
   FILE* fp = fopen(name, "w");
   if (fp == (FILE *) 0) fprintf(stderr, "Could not open %s for writing", name);
   return fp;
}

void output(int* hist) {
   char filename [1024];
   sprintf(filename, "angular_correlation.res");
   FILE* fp = openfile(filename);

   unsigned int sum=0;
   for (int i=0; i<nbins; i++) sum += hist[i];
   for (int i=0; i<nbins; i++) fprintf(fp, "%e - %e : %e\n", -1+2.*i/nbins, -1+2.*(i+1)/nbins, 1.*hist[i]/sum);

   fclose(fp);
   return;
}
```

In the initialization phase, the random number generator is seeded, the call to **fissionEvent::setCorrelationOption(3)** turns on FREYA. A new instance of class fissionEvent is created for each new fission, from which we extract the directions of the emitted neutrons. Potential error messages produced by FREYA can be retrieved by the programmer via a call to **fissionEvent::getFREYAerrors()**. An example of such error is failure to specify a valid location for FREYA's data using **FREYADATAPATH**.

This code produces an output file with the distribution of angles between fission neutrons. This distribution is shown in Fig. 8.

C Code using the C API

The second code “angular_correlation.c” interfaces with the LLNL Fission Library using the C interface described in Sec. A.2.

```
#include "Fission.h"

int main() {
   int isotope = 94239;
   double energy_MeV = 2.;
   double nubar = 3.163;
   double time = 0.;

   int maxerrorlength=10000;
   char errors[maxerrorlength];

   int i, hist[nbins];
   for (i=0; i<nbins; i++) hist[i] = 0.;

   init();
   for (i=0; i<iterations; i++) {
      genfissevt_(&isotope, &time, &nubar, &energy_MeV);
      int errorlength=maxerrorlength;
      getfreya_errors_(&errorlength, &errors[0]);
      if (errorlength>1) {
        printf("%s\n",errors);
        exit(1);
      }
      int nneutrons = getnnu_();
      int n1;
      for(n1=0; n1<nneutrons; n1++) {
         double u1 = getndircosu_(&n1), v1 = getndircosv_(&n1), w1 = getndircosw_(&n1);
         int n2;
         for(n2=n1+1; n2<nneutrons; n2++) {
            double u2 = getndircosu_(&n2), v2 = getndircosv_(&n2), w2 = getndircosw_(&n2);
            double scalar_product = u1*u2+v1*v2+w1*w2;

            int bin_index = (int) (nbins*(scalar_product+1)/2);
            hist[bin_index]++;
         }
      }
   }
   output(hist);
}

void init(void) {
   unsigned short int s[3] = {1234, 5678, 9012};
   int i, three = 3;
   seed48(s);
   setcorrel_(&three);
   return;
}
```

Some sections from this code were removed because they are identical to those presented in Sec. B.

References

[1] X-5 Monte Carlo Team, "MCNP – A General Monte Carlo N-Particle Transport Code, Version 5 - Volume I: Overview and Theory," Los Alamos National Laboratory, Los Alamos, NM, LA-UR-03-1987 (2008). 3

[2] X-5 Monte Carlo Team, "MCNP – A General Monte Carlo N-Particle Transport Code, Version 5 - Volume II: User's Guide," Los Alamos National Laboratory, Los Alamos, NM, LA-CP-03-0245 (2003). 3

[3] X-5 Monte Carlo Team, "MCNP – A General Monte Carlo N-Particle Transport Code, Version 5 - Volume III: Developer's Guide," Los Alamos National Laboratory, Los Alamos, NM, LA-CP-03-0284 (2003). 3

[4] T. Goorley *et al.* "Initial MCNP6 Release Overview - MCNP6 version 1.0," Los Alamos National Laboratory, Los Alamos, NM, LA-UR-13-22934 (2013). 3

[5] Available from the Radiation Safety Information Computational Center, http://rsicc.ornl.gov. 3

[6] D. E. Cullen, "TART2005: A Coupled Neutron-Photon 3-D, Combinatorial Geometry, Time Dependent Monte Carlo Transport Code," Lawrence Livermore National Laboratory, Livermore CA, UCRL-SM-218009 (2005). 3

[7] R. M. Buck and E. M. Lent, "COG User's Manual: A Multiparticle Monte Carlo Transport Code," Lawrence Livermore National Laboratory, Livermore CA, UCRL-TM-202590, 5th Edition (2002). 3

[8] S. Agostinelli et al., "GEANT4 - a simulation toolkit," Nucl. Instr. Meth. Phys. Res. A **506** (2003) 250-303. 3

[9] J. Allison et al., "Geant4 developments and applications," IEEE Trans. Nucl. Sci. **53** No. 1 (2006) 270-278. 3

[10] Available from the European Organization for Nuclear Research, http://geant4.cern.ch. 3

[11] T. E. Valentine, "MCNP-DSP Users Manual," Oak Ridge National Laboratory, Oak Ridge, TN, ORNL/TM-13334 R2 (January 2001). 3

[12] E. Padovani, S. A. Pozzi, S. D. Clarke and E. C. Miller, "MCNPX-PoliMi User's Manual," C00791 MNYCP, Radiation Safety Information Computational Center, Oak Ridge National Laboratory (2012). 3

[13] D. B. Pelowitz *et al.*, "MCNPX2.7.0 Extensions," Los Alamos National Laboratory, Los Alamos, NM, LA-UR-11-02295 (2011). 3

[14] J. M. Verbeke, C. Hagmann and D. Wright, "Simulation of Neutron and Gamma-Ray Emission from Fission and Photofission," Lawrence Livermore National Laboratory, Livermore CA, UCRL-AR-228518 (2010). 3, 25, 35

[15] S. Lemaire, P. Talou, T. Kawano, M. B. Chadwick and D. G. Madland, Phys. Rev. C **72** (2005) 024601. 3, 10

[16] S. Lemaire, P. Talou, T. Kawano, M. B. Chadwick and D. G. Madland, Phys. Rev. C **73** (2006) 014602. 3

[17] J. Randrup and R. Vogt, Phys. Rev. C **80** (2009) 024601. 3, 8

[18] R. Vogt, J. Randrup, J. Pruet and W. Younes, Phys. Rev. C **80** (2009) 044611. 3

[19] R. Vogt and J. Randrup, Phys. Rev. C **84** (2011) 044612. 3, 8

[20] R. Vogt, J. Randrup, D. A. Brown, M. A. Descalle and W. E. Ormand, Phys. Rev. C **85** (2012) 024608. 3, 4, 5, 7, 8, 10, 19, 20, 24

[21] R. Vogt and J. Randrup, Phys. Rev. C **87** (2013) 044602. 3

[22] J. Randrup and R. Vogt, Phys. Rev. C **89** (2014) 044601. 3

[23] TRIPOLI-4® Project Team: "TRIPOLI-4 version 8 User Guide," CEA-R-6316, Feb. 2013. 3

[24] W. J. Swiatecki, K. Siwek-Wilczyńska and J. Wilczyński, Phys. Rev. C **78** (2008) 054604. 6, 15

[25] W. Younes *et al*, Phys. Rev. C **64** (2001) 054613. 8

[26] W. Reisdorf, J. P. Unik, H. C. Griffin and L. E. Glendenin, Nucl. Phys. A **177** (1971) 337. 8

[27] G. Audi and A. H. Wapstra, Nucl. Phys. A **595** (1995) 409. 8, 22

[28] P. Möller, J. R. Nix, W. D. Myers and W. J. Swiatecki, Atomic Data and Nucl. Data Tab. **59** (1995) 185. 8, 22, 24

[29] T. Kawano, S. Chiba and H. Koura, J. Nucl. Sci. Technol. **43** (2006) 1. 10

[30] H. Koura, M. Uno, T. Tachibana and M. Yamada, Nucl. Phys. A **674** (2000) 47. 10, 24

[31] C. Hagmann, J. Randrup and R. Vogt, Trans. Nucl. Sci. **60** (2013) 545. 26

[32] International Handbook of Evaluated Criticality Safety Benchmark Experiments, NEA Nuclear Science Committee (2007), http://icbep.inel.gov. 26

[33] J. M. Verbeke, C. A. Hagmann, J. Randrup and R. Vogt, "Integration of FREYA into MCNP6: An Improved Fission Chain Modeling Capability," Lawrence Livermore National Laboratory, LLNL-PROC-638986 (2013). 26, 34

[34] S. A. Pozzi *et al*, Nucl. Sci. Eng. **178** (2014) 1. 26, 34

[35] R. Vogt and J. Randrup, "Neutron angular correlations in spontaneous and neutron-induced fission", Phys. Rev. C **90** (2014) 064623. 26, 34

[36] W. H. Press, S. A. Teukolsky, W. T. Vetterling, B. P. Flannery, "Numerical Recipes 3rd Edition: The Art of Scientific Computing," Cambridge University Press New York, NY, USA (2007). 26

[37] B. Beck, D. A. Brown, F. Daffin, J. Hedstrom and R. Vogt, "Implementation of Energy-Dependent Q Values for Fission," UCRL-TR-234617, Lawrence Livermore National Laboratory (2007). 36

[38] R. Vogt, "Energy-Dependent Fission Q Values Generalized for All Actinides," LLNL-TR-407620, Lawrence Livermore National Laboratory (2008). 36

[39] M. S. Zucker and N. E. Holden, "Energy Dependence of Neutron Multiplicity $P(\nu)$ in Fast-Neutron-Induced Fission for ^{235,238}U and ^{239}Pu," BNL-38491 (1986). 36

[40] R. Gwin, R. R. Spencer and R. W. Ingle, Nucl. Sci. Eng. **87** (1984) 381. 36

[41] R. R. Spencer, R. Gwin and R. W. Ingle, Nucl. Sci. Eng. **80** (1982) 603. 37

[42] J. W. Boldeman and M. G. Hines, Nucl. Sci. Eng. **91** (1985) 114. 37

[43] W. Mannhart, “Evaluation of the ^{252}Cf Fission Neutron Spectrum Between 0 MeV and 20 MeV,” Proc. Advisory Group Mtg. Neutron Sources, Leningrad, USSR, 1986 (IAEA-TECDOC-410), Vienna (1987). 37

[44] D. G. Madland and J. R. Nix, Nucl. Sci. Eng. **81** (1982) 213. 37

[45] B. E. Watt, Phys. Rev. **87** (1952) 1037. 37

[46] F. H. Fröhner, Nucl. Sci. Eng. **106** (1990) 345. 37